AL/CF-TR-1994-0047-Vol. 2

AD-A281 932

DTIC
ELECTE
JUL 0 7 1994
S B D

ARMSTRONG LABORATORY

MODELING RESPIRATORY GAS DYNAMICS IN THE AVIATOR'S BREATHING SYSTEM

John B. Bomar, Jr.
Michael W. Scott
Darrin A. Smith

Biodynamic Research Corporation
9901 IH 10 West, Suite 1000
San Antonio, TX 78230

CREW SYSTEMS DIRECTORATE
CREW TECHNOLOGY DIVISION
2504 D Drive, Suite 1
Brooks Air Force Base, TX 78235-5104

413 87

94-20738

May 1994

Final Technical Report for Period May 1993 - December 1993

Approved for public release; distribution is unlimited.

132

DTIC QUALITY INSPECTED 5

AIR FORCE MATERIEL COMMAND
BROOKS AIR FORCE BASE, TEXAS

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF COLOR PAGES WHICH DO NOT REPRODUCE LEGIBLY ON BLACK AND WHITE MICROFICHE.

NOTICES

This research was conducted under the Small Business Innovation Research (SBIR) Program as a Phase I effort.

Publication of this report does not constitute approval or disapproval of the ideas or findings. It is published in the interest of STINFO exchange.

The U.S. Government does not require SBIR Phase I contractors to adhere to any particular format or style. In accordance with SBIR guidelines for Phase I efforts, the contractor's report is accepted for publication but is not edited.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Customers can obtain a copy of the Appendixes, Volume 2, by writing to the Defense Technical Information Center (DTIC), Building #5, Cameron Station, 5010 Duke Street, Alexandria, VA 22304-6145.

The Office of Public Affairs has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for publication.

Robert B. O'Connor

ROBERT B. O'CONNOR, Captain, USAF
Project Scientist

F.W. B

F. WESLEY BAUMGARDNER, Ph.D.
Chief, Systems Research Division

Ronald C. Hill

RONALD C. HILL, Colonel, USAF, BSC
Acting Chief, Crew Technology Division

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification

By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
May 1994

3. REPORT TYPE AND DATES COVERED
Final May 1993 - December 1993

4. TITLE AND SUBTITLE
Modeling Respiratory Gas Dynamics in the Aviator's Breathing System

5. FUNDING NUMBERS
C - F41624-93-C-6009
PE - 65502F
PR - 3005
TA - CT
WU - 3A

6. AUTHOR(S)
John B. Bomar, Jr.
Michael W. Scott
Darrin A. Smith

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Biodynamic Research Corporation
9901 IH 10 West, Suite 1000
San Antonio, TX 78230

8. PERFORMING ORGANIZATION REPORT NUMBER
F41624-93-C-6009

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Armstrong Laboratory (AFMC)
Crew Systems Directorate
Crew Technology Division
2504 D Drive, Suite 1
Brooks Air Force Base, TX 78235-5104

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AL/CF-TR-1994-0047-Vol. 2

11. SUPPLEMENTARY NOTES
Armstrong Laboratory Technical Monitor: Captain Robert B. O'Connor, (210) 536-3545. This research was conducted under the Small Business Innovation Research (SBIR) program as a Phase I effort.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Biodynamic Research Corporation (BRC) completed an SBIR Phase I project to study the feasibility of developing a model of the Aviator's Breathing System (ABS). The motivation for the project was the desire to develop a model which could simulate the cardiovascular and respiratory responses to altitude and acceleration stress encountered in high performance military aircraft. Software modules were developed and tested for simulation of: (1) the flows and pressures within the breathing gas delivery system; (2) the flows, pressures, and gas distribution within the lung; and (3) the steady-state flows and pressures within the cardiovascular system. Subprograms were also developed to compute altitude barometric pressure relationships as well as passenger cabin pressures in military aircraft. In addition to the software development, BRC reviewed and organized the Government furnished data from a series of manned rapid decompression known as the EONS Experiments. The data from approximately 170 experimental decompressions were screened for their suitability for use in parameter selection and validation of the respiratory modeling software. The data appears to be highly coherent and fully usable for model validation. We conclude that the development of an integrated ABS Model is feasible and desirable.

14. SUBJECT TERMS
Aircrew breathing system
Lung mechanics
Mathematical model
Rapid decompression
Respiratory physiology

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

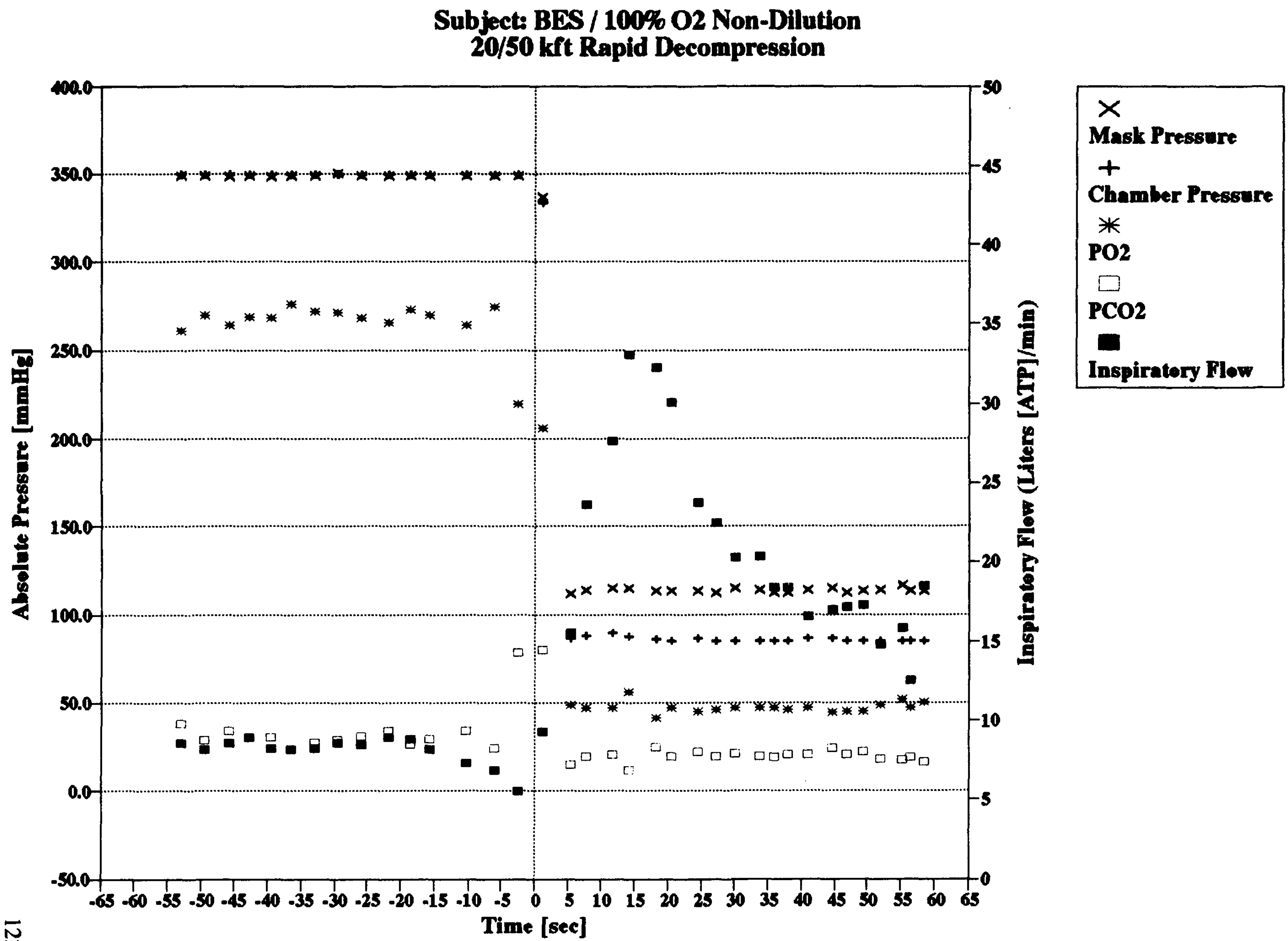
NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std Z39-18
298-102

APPENDICES

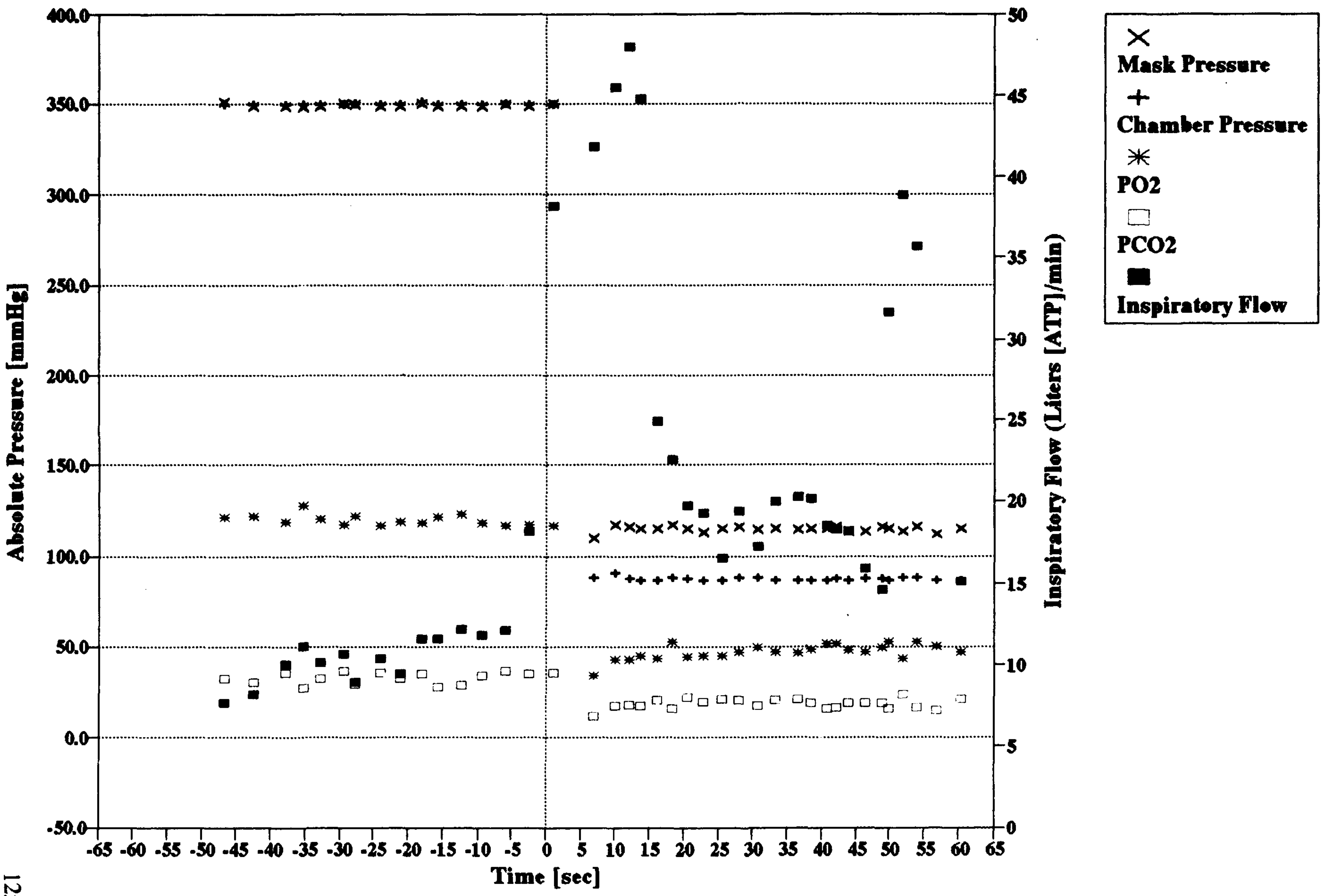
APPENDIX A

Rapid Decompression Data Plots

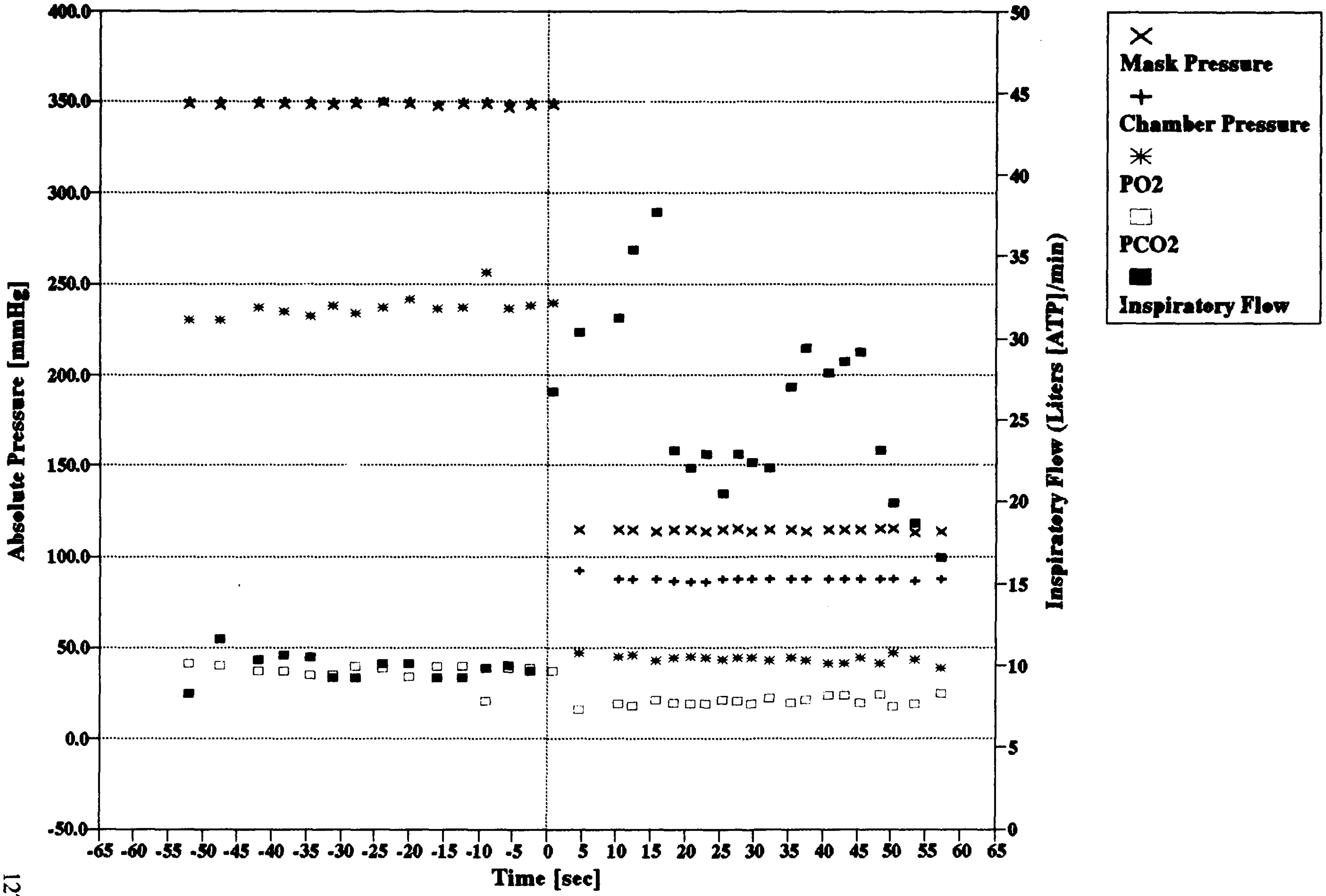


Subject: BES / 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

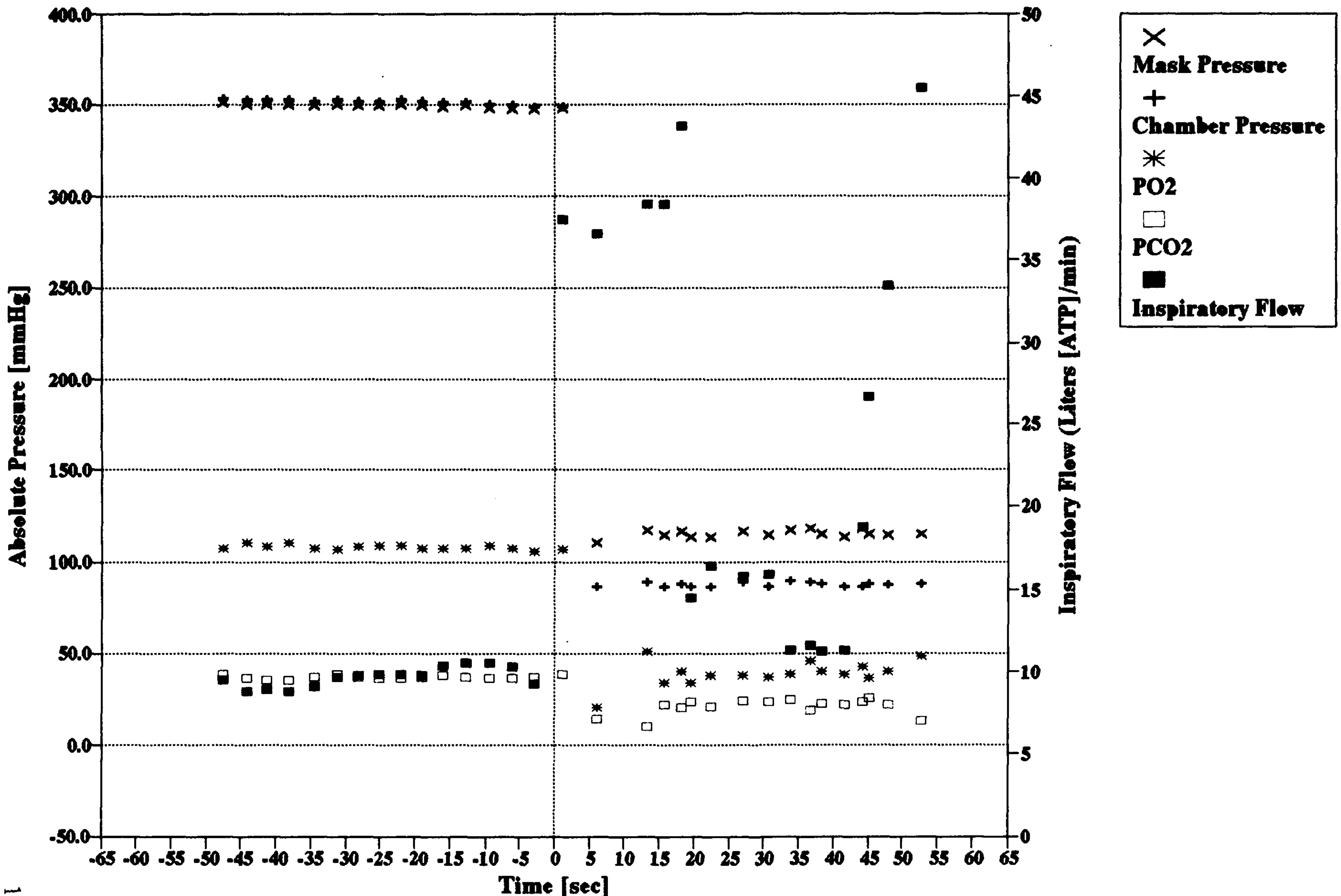
Subject: BES / 100% O2 Dilution
20/50 kft Rapid Decompression



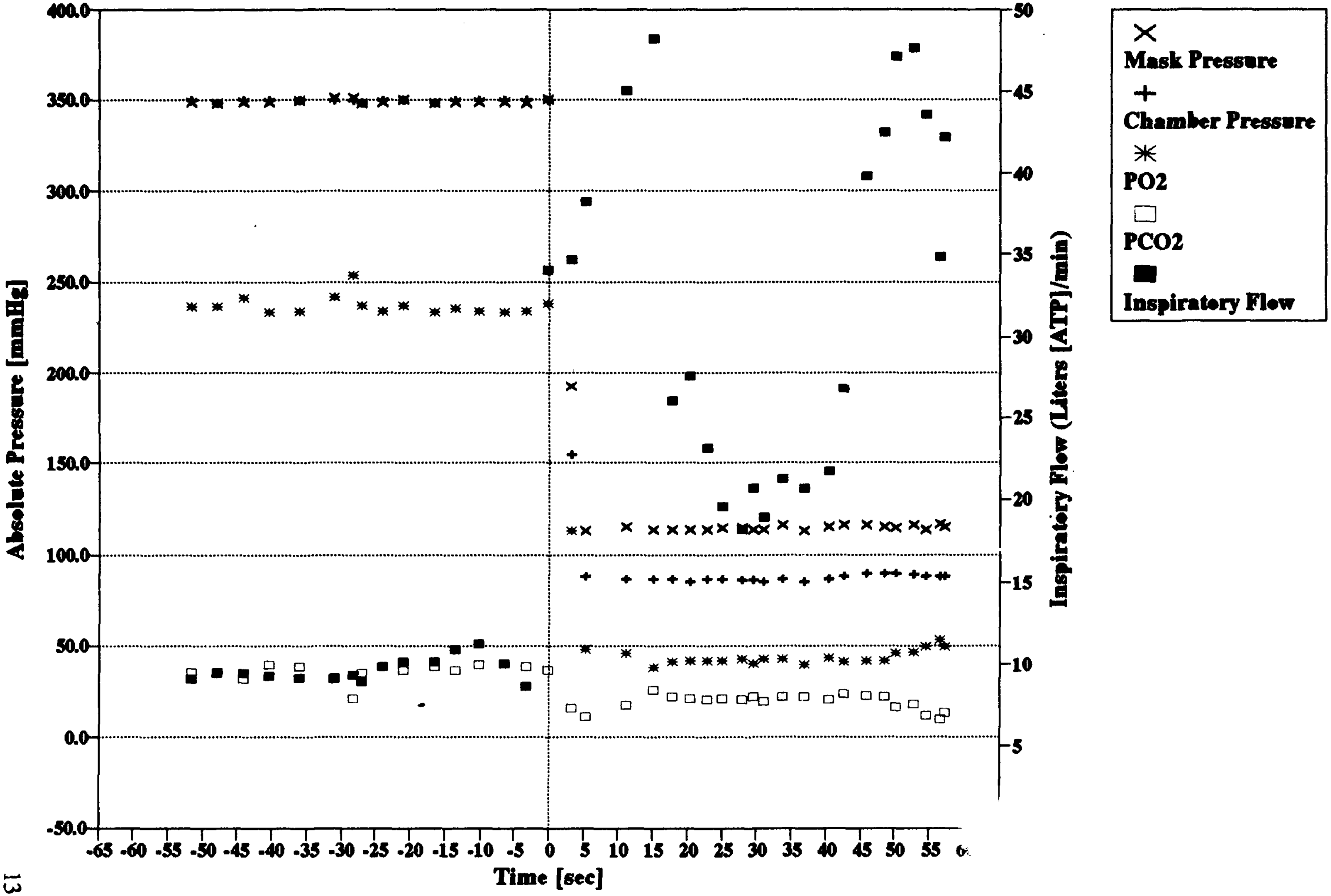
Subject: BES / 93% O2 Non-Dilution
20/50 kft Rapid Decompression

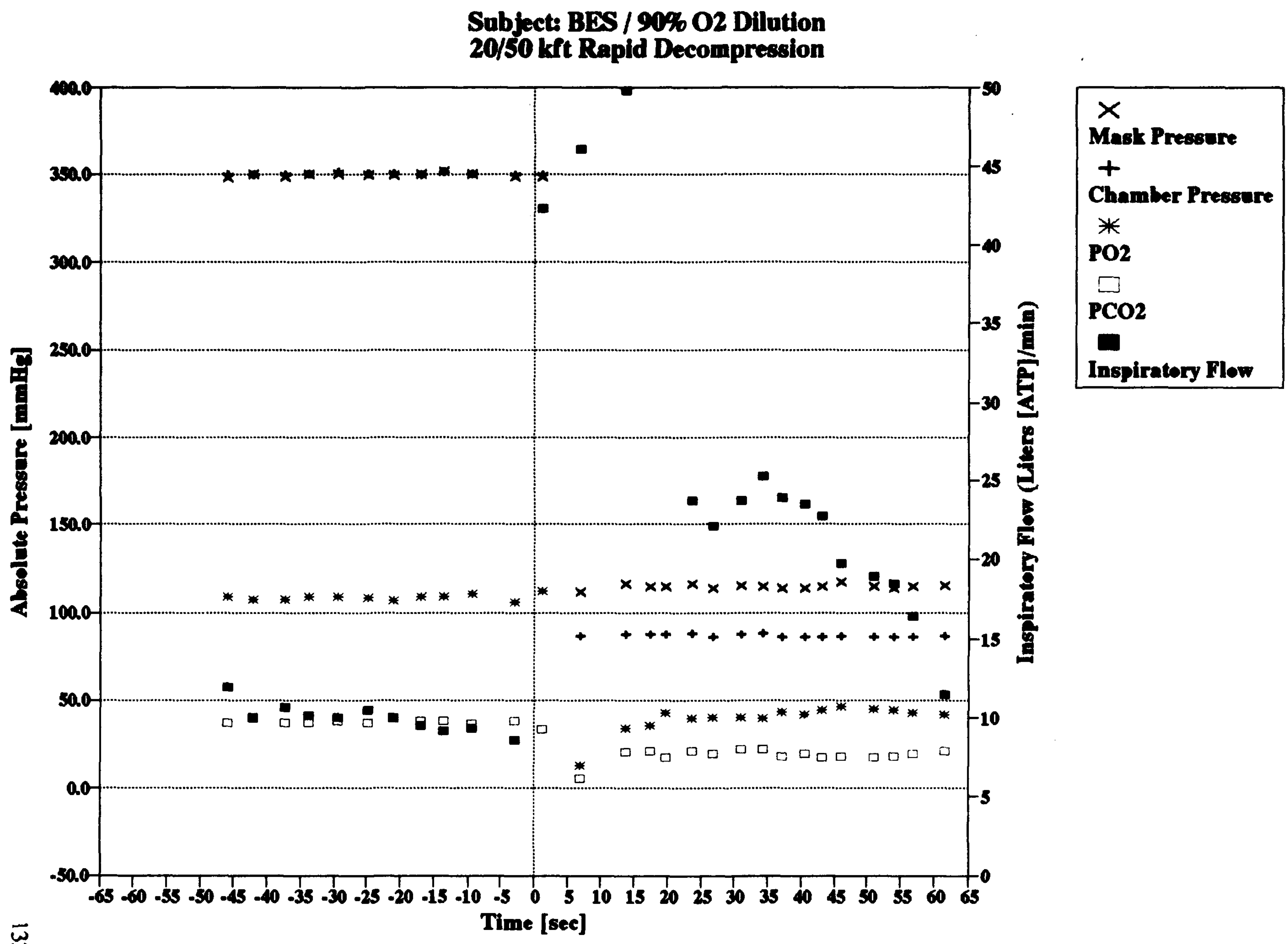


Subject: BES / 93% O2 Dilution
20/50 kft Rapid Decompression

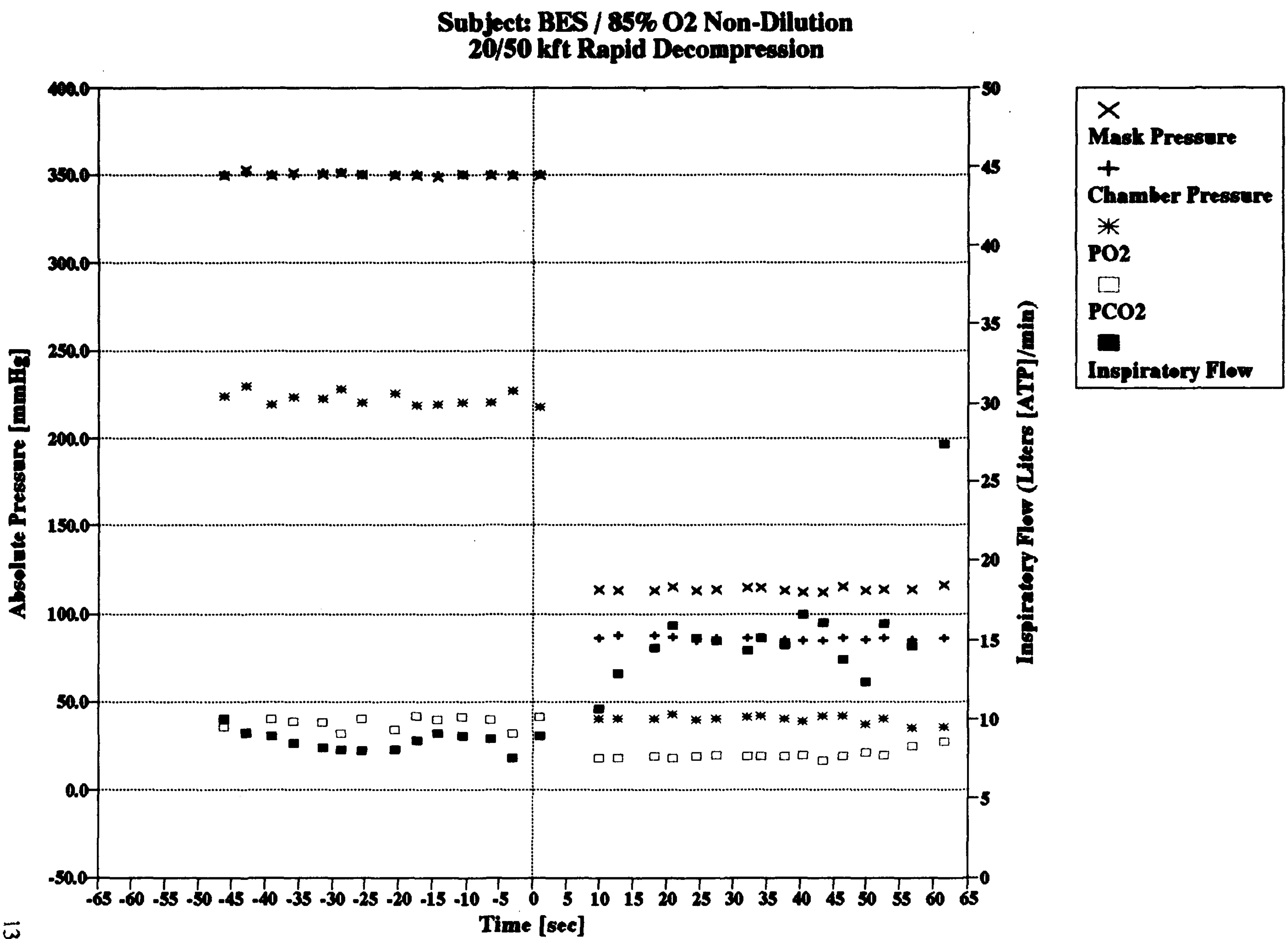


Subject: BES / 90% O2 Non-Dilution
20/50 kft Rapid Decompression



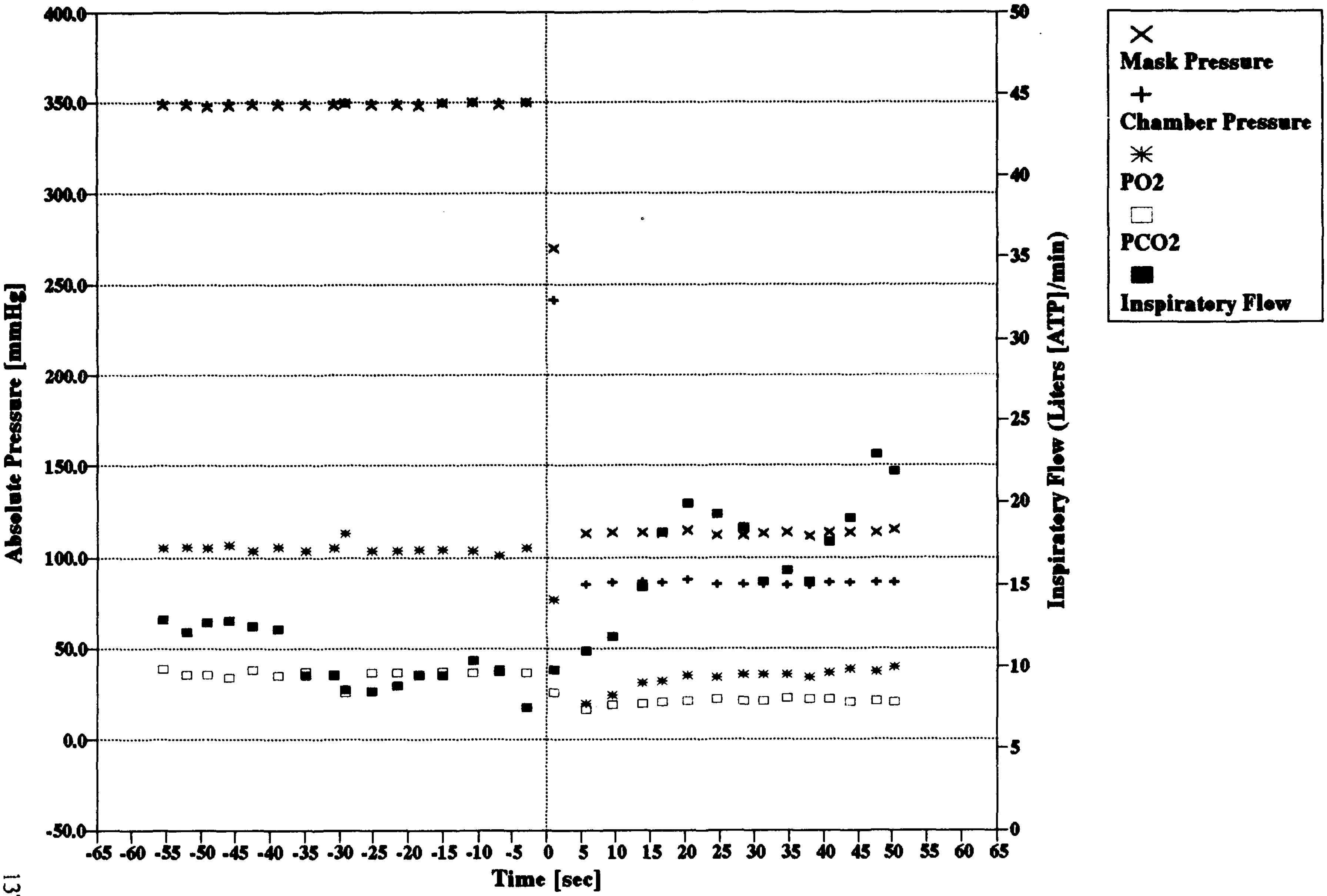


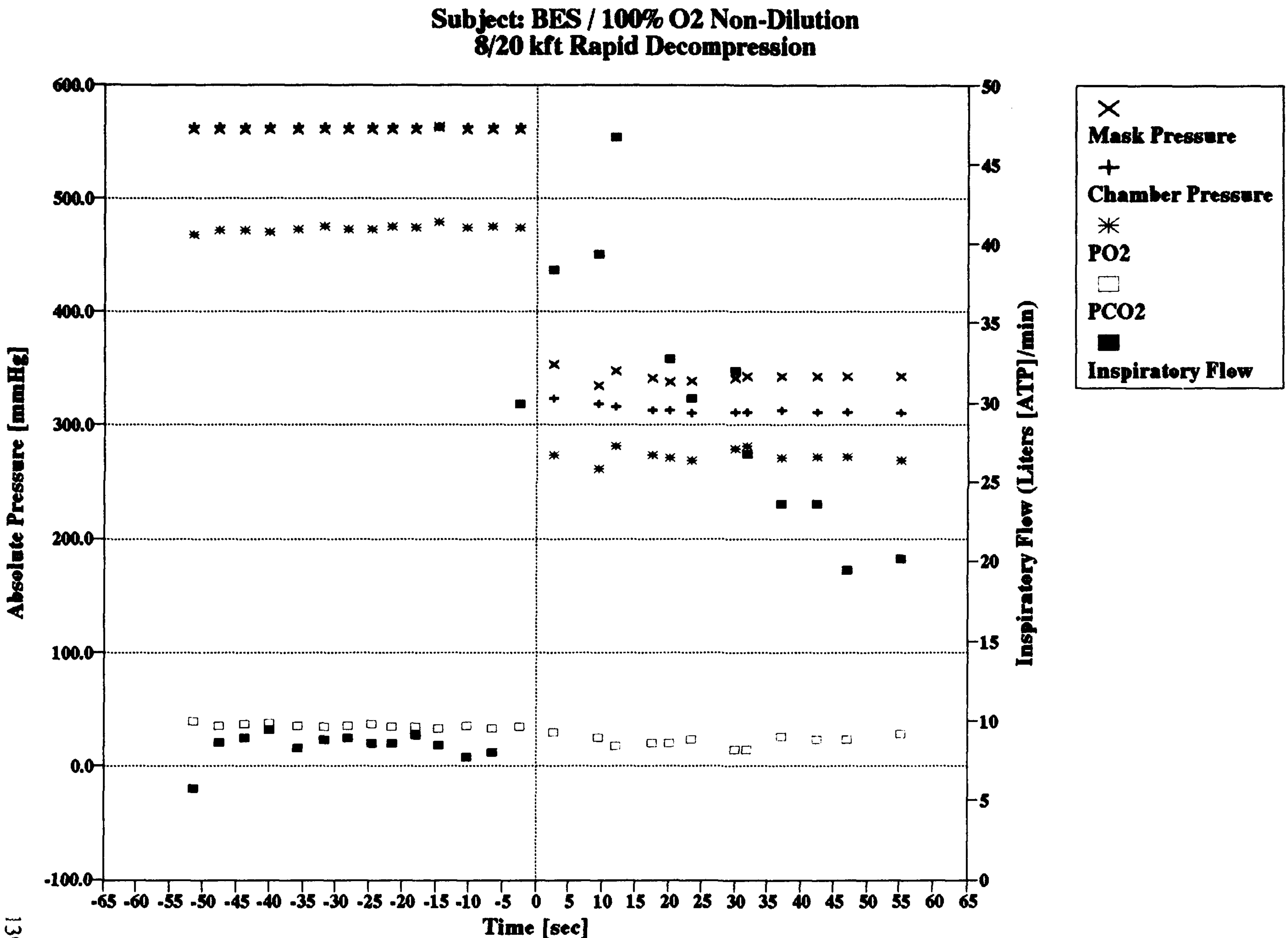
Subject: BES / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow



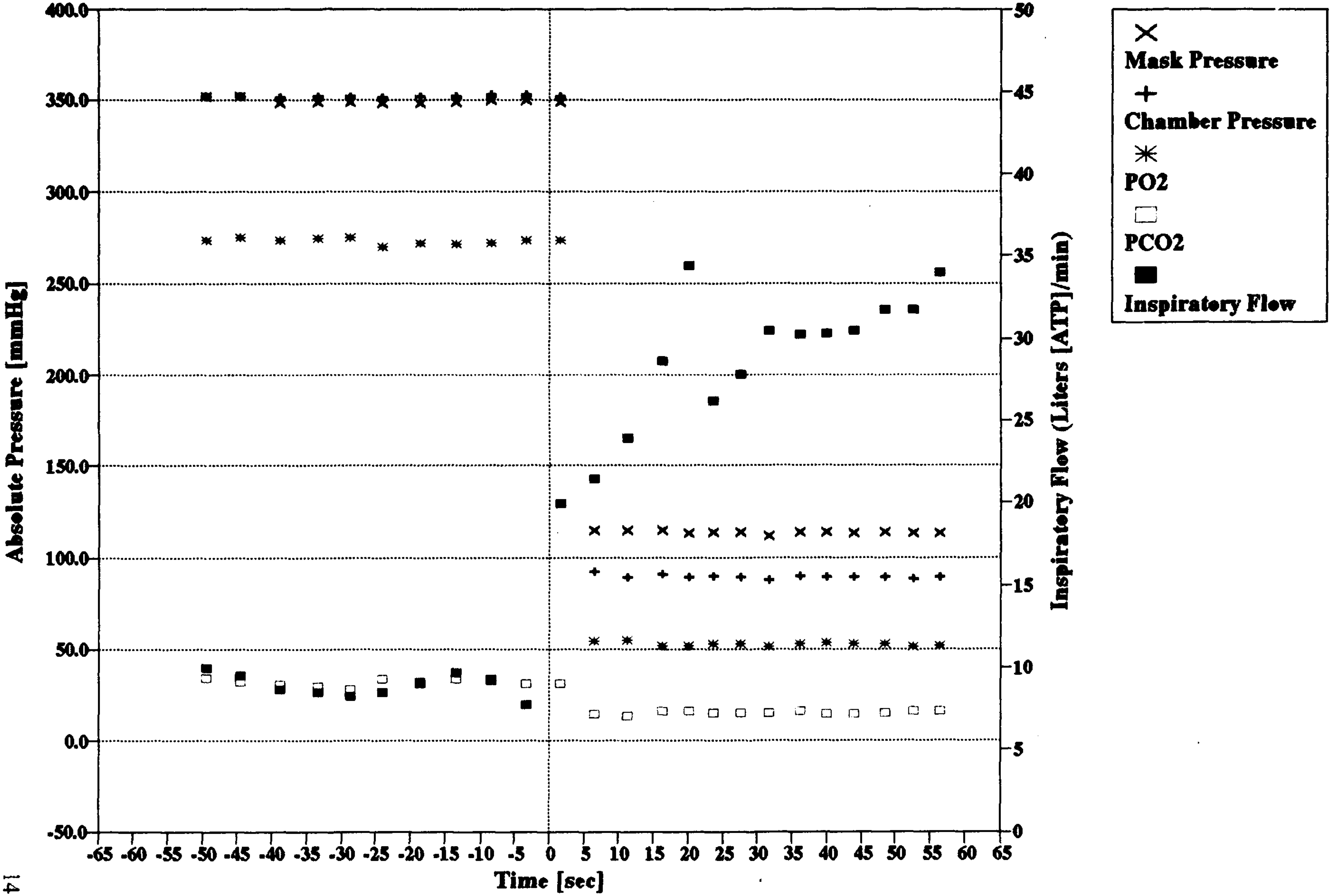
Subject: BES / 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: BES / 85% O2 Dilution
20/50 kft Rapid Decompression

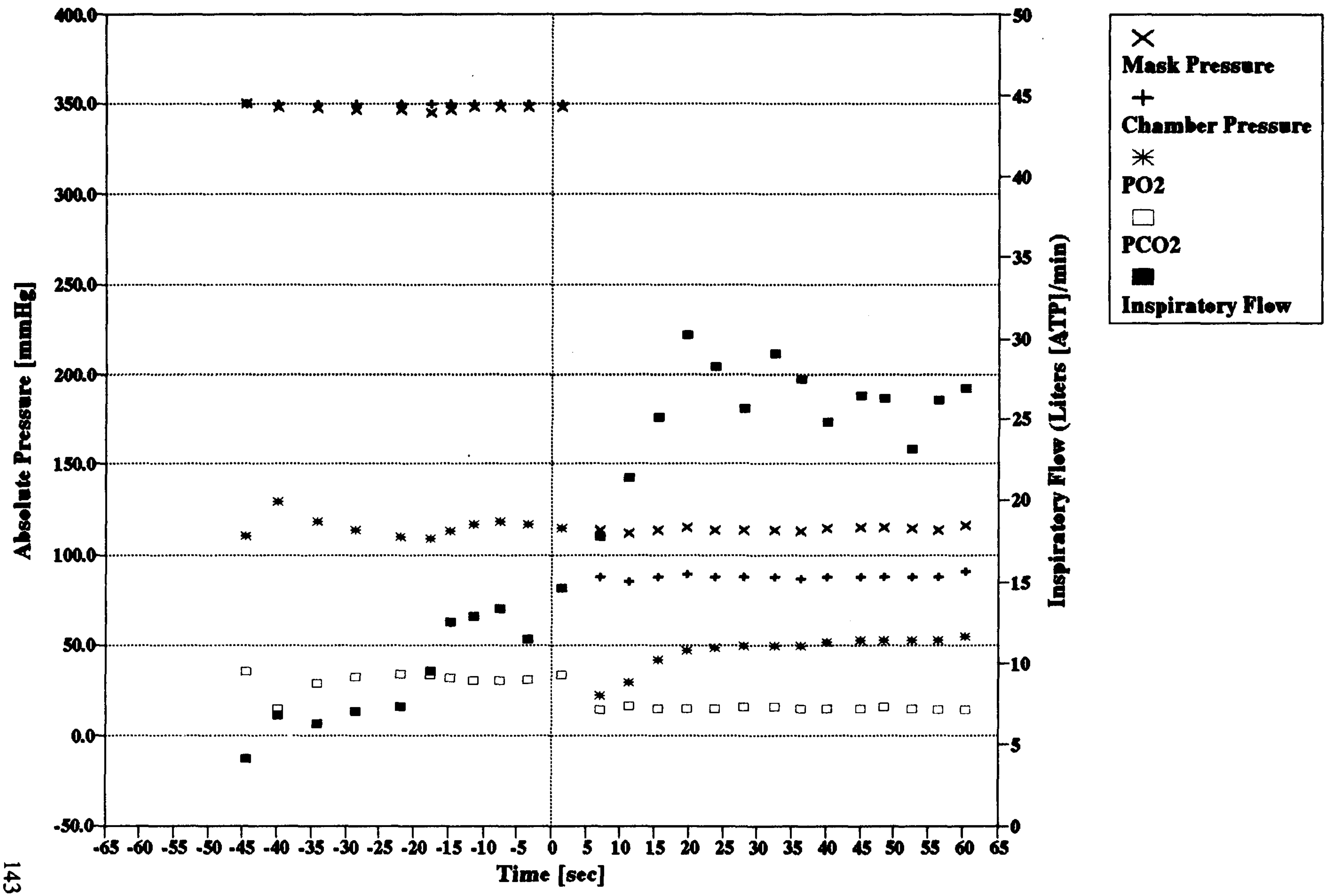


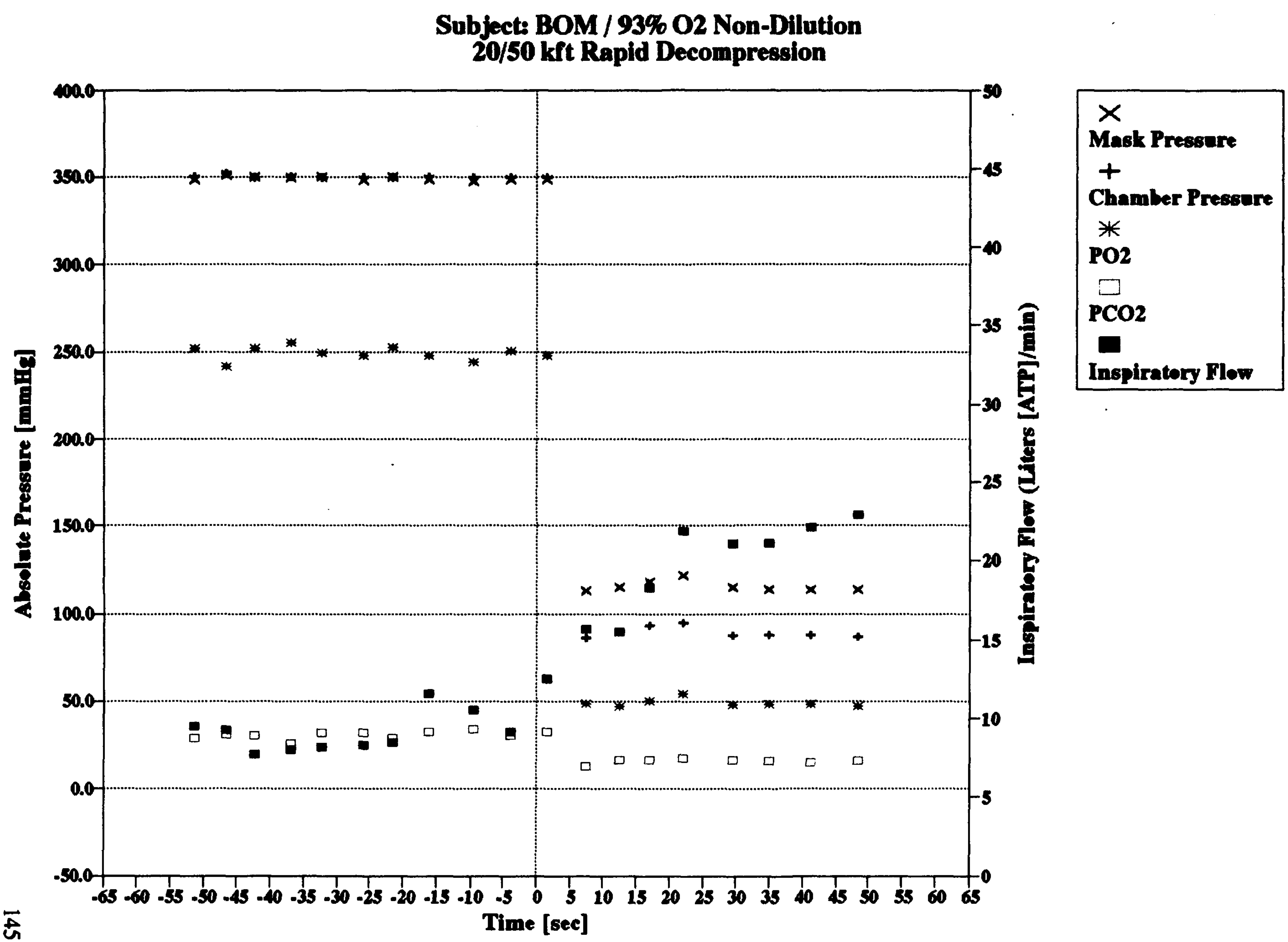

Subject: BES / 100% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: BOM / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



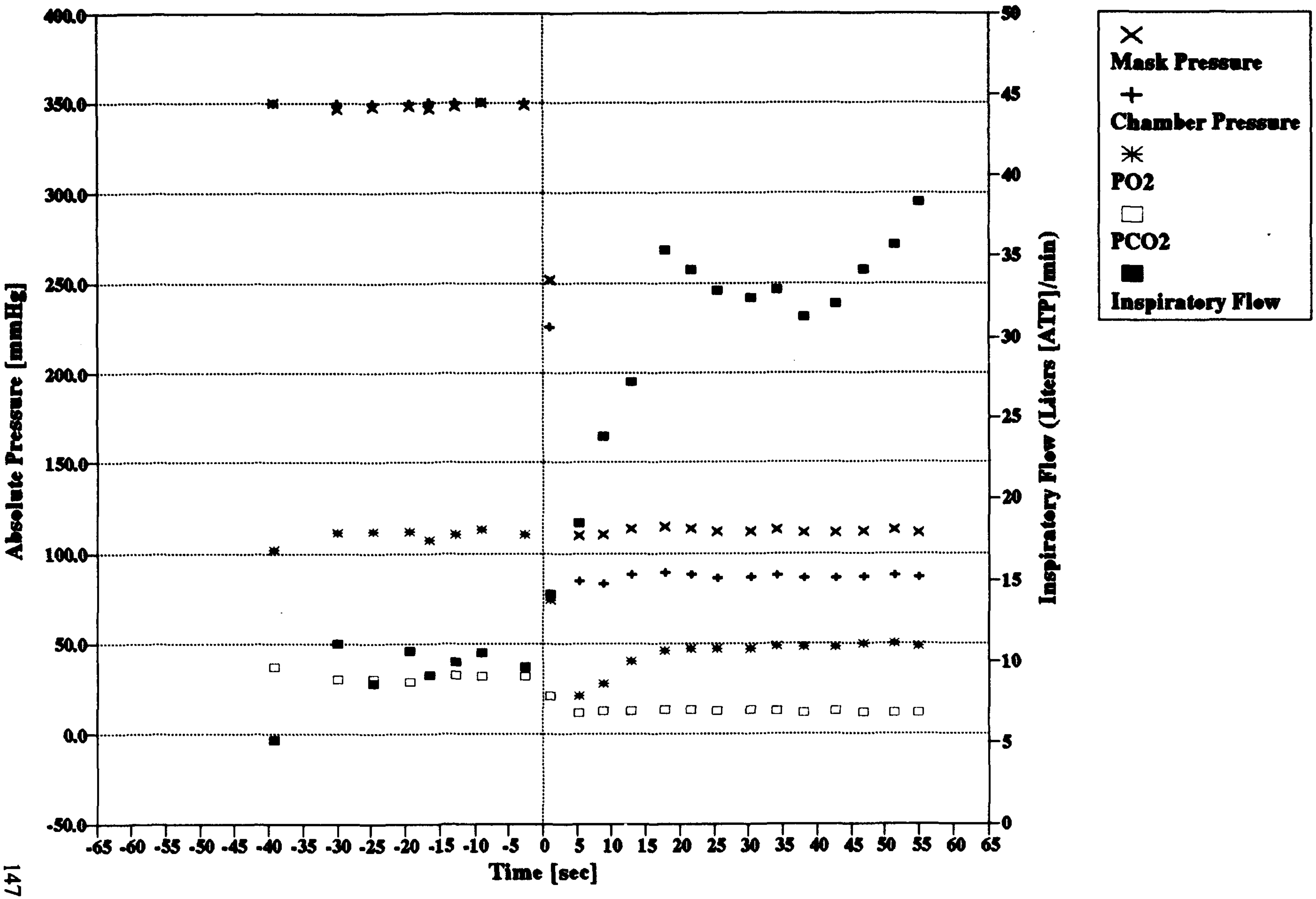
Subject: BOM / 100% O2 Dilution
20/50 kft Rapid Decompression



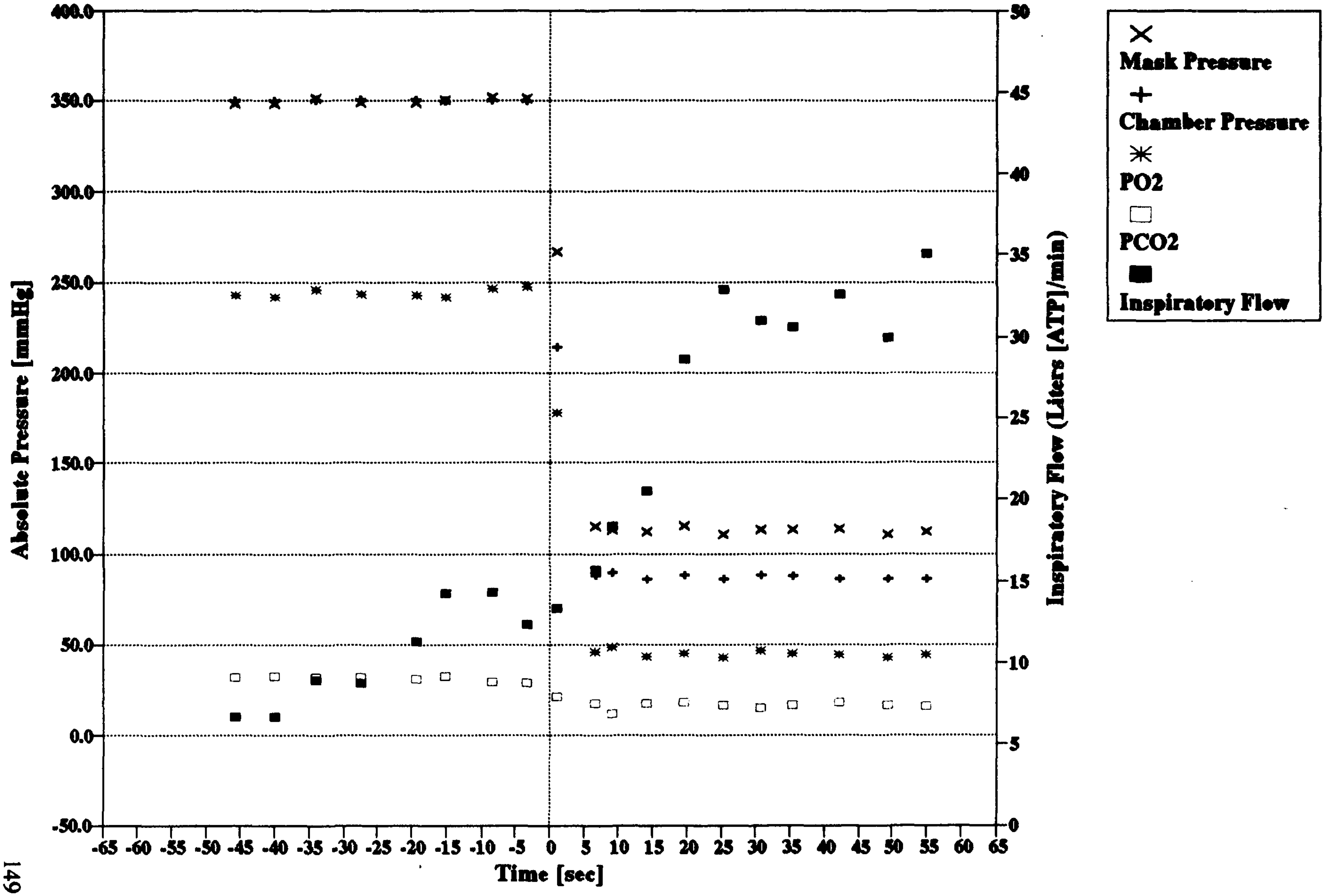


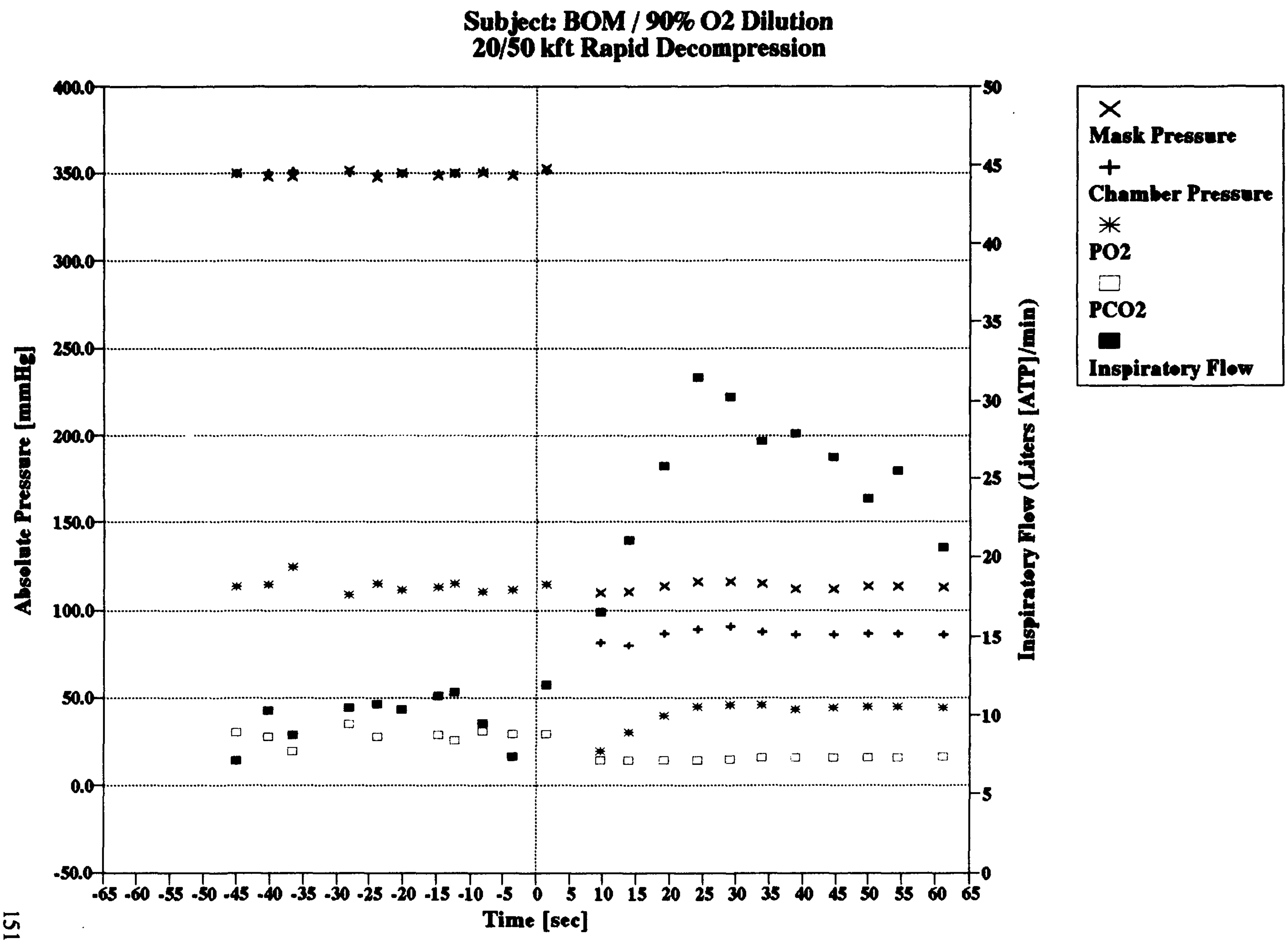
Subject: BOM / 93% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: BOM / 93% O2 Dilution
20/50 kft Rapid Decompression

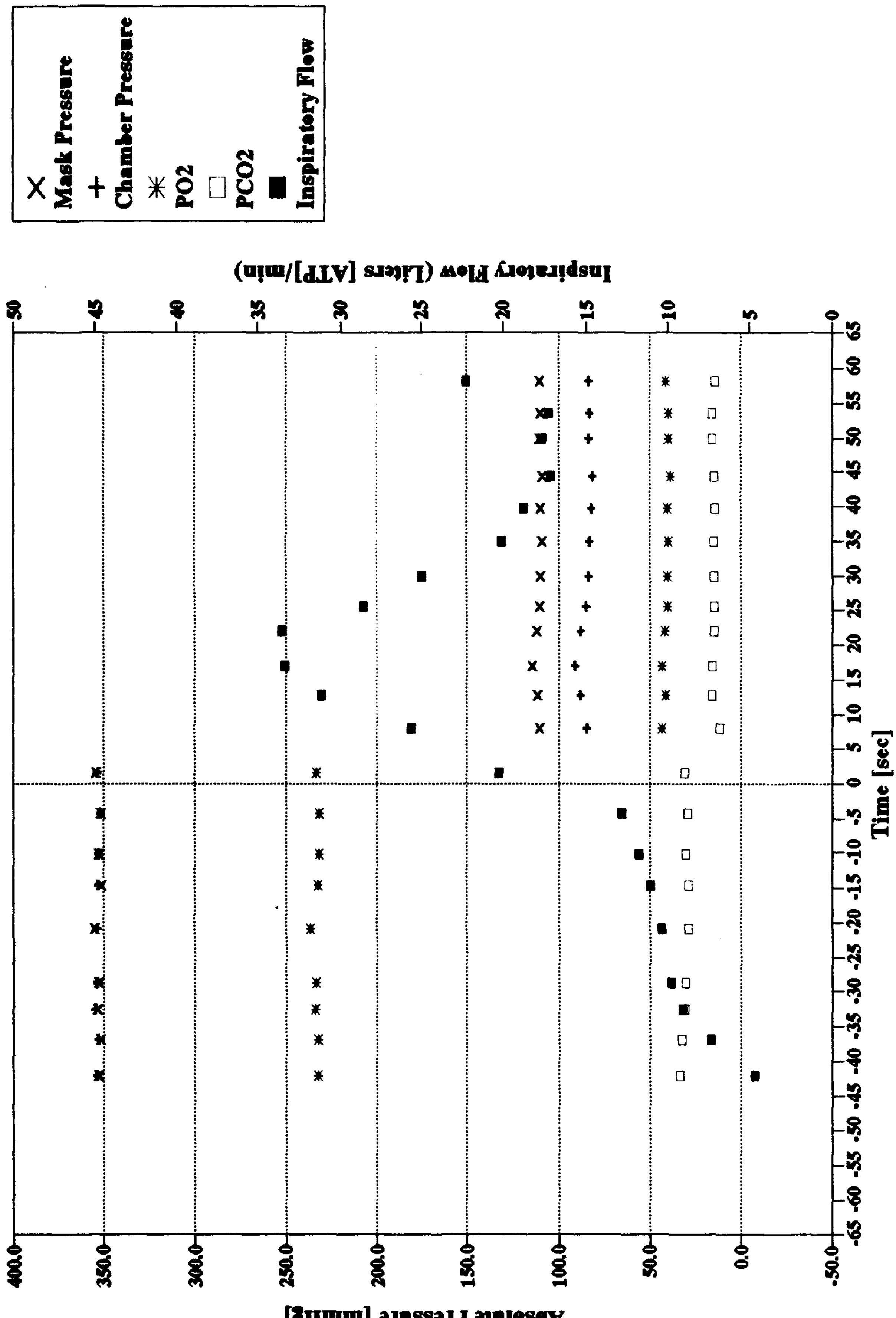


Subject: BOM / 90% O2 Non-Dilution
20/50 kft Rapid Decompression

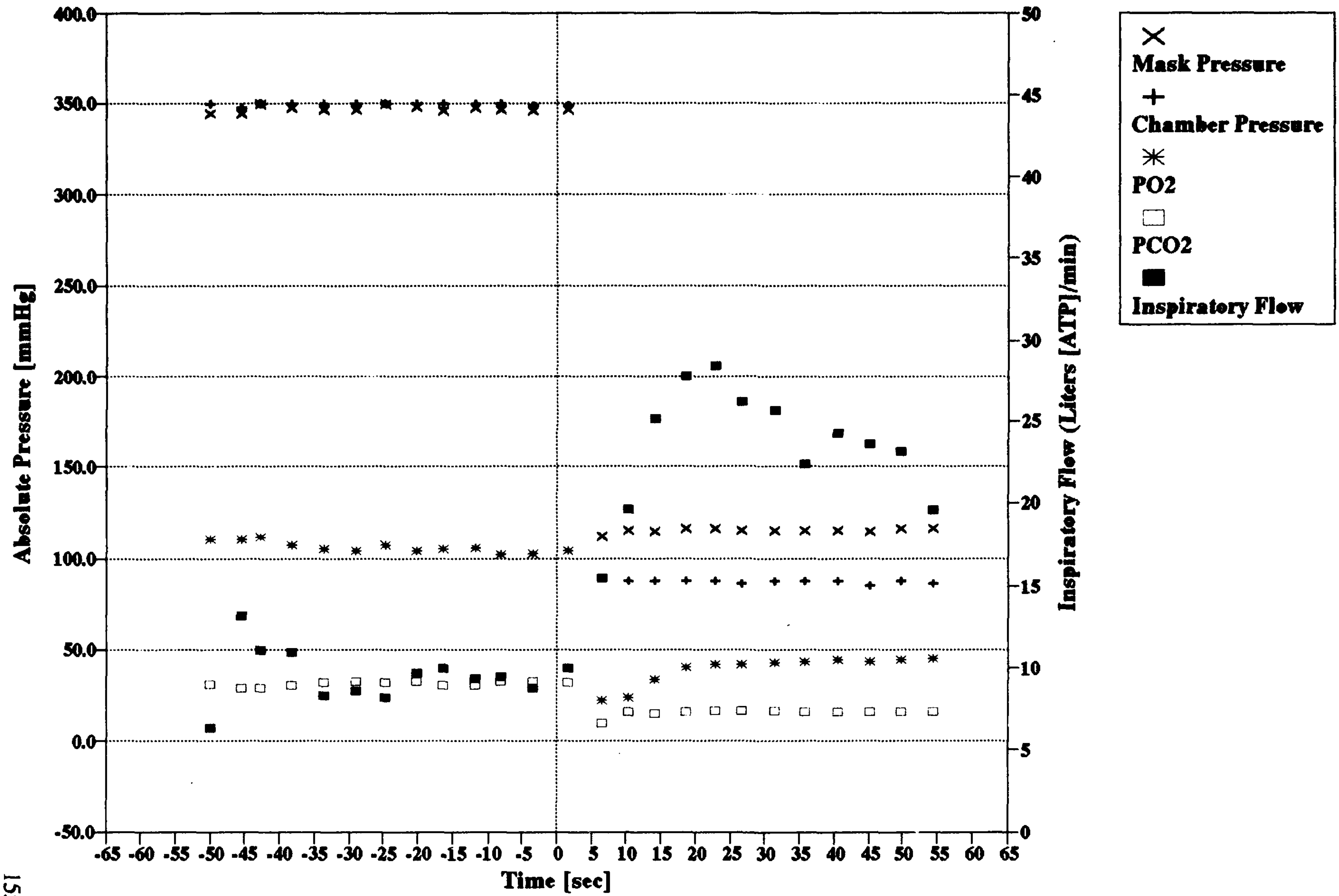



Subject: BOM / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

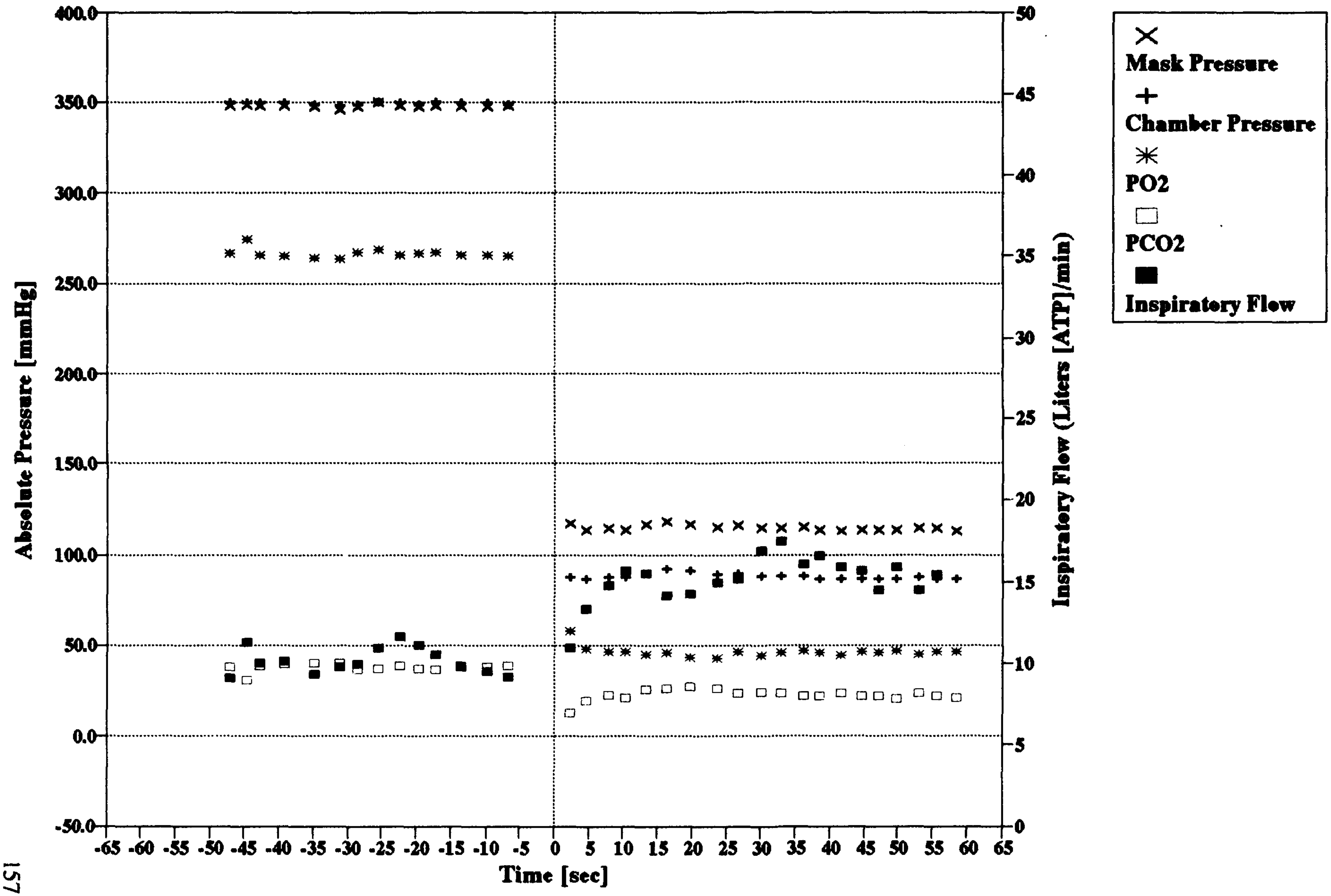
Subject: BOM / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



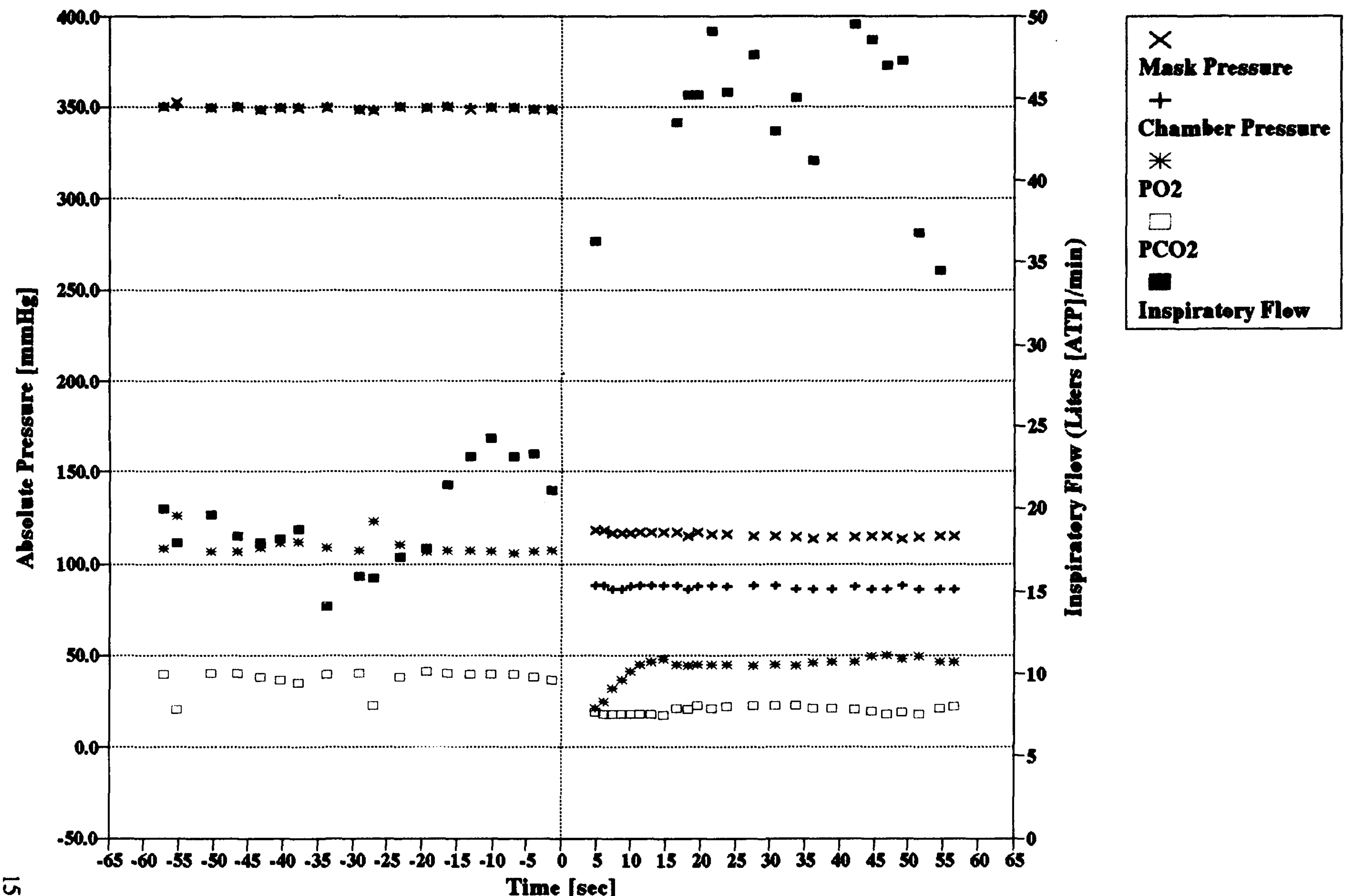
Subject: BOM / 85% O2 Dilution
20/50 kft Rapid Decompression



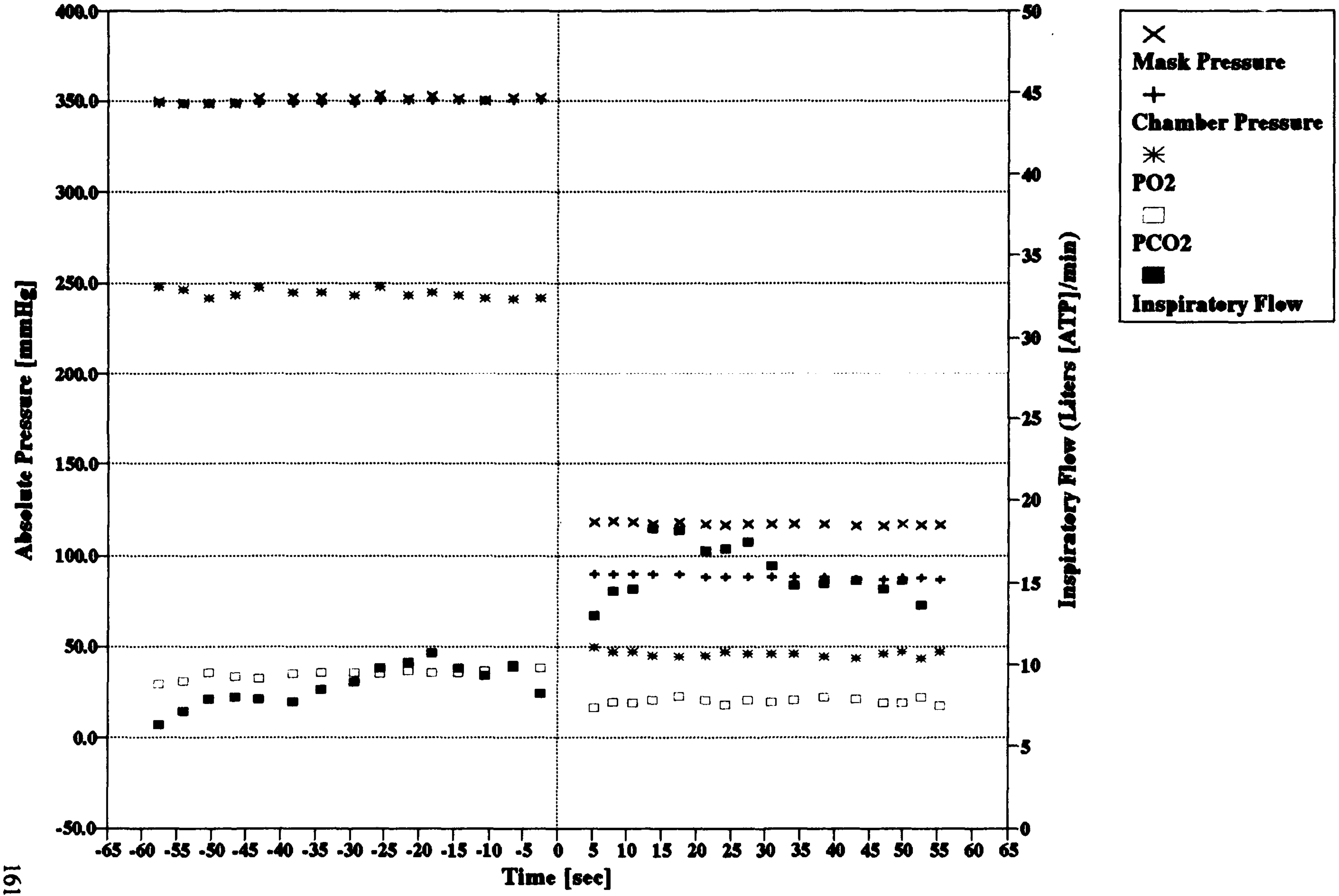
Subject: COX / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



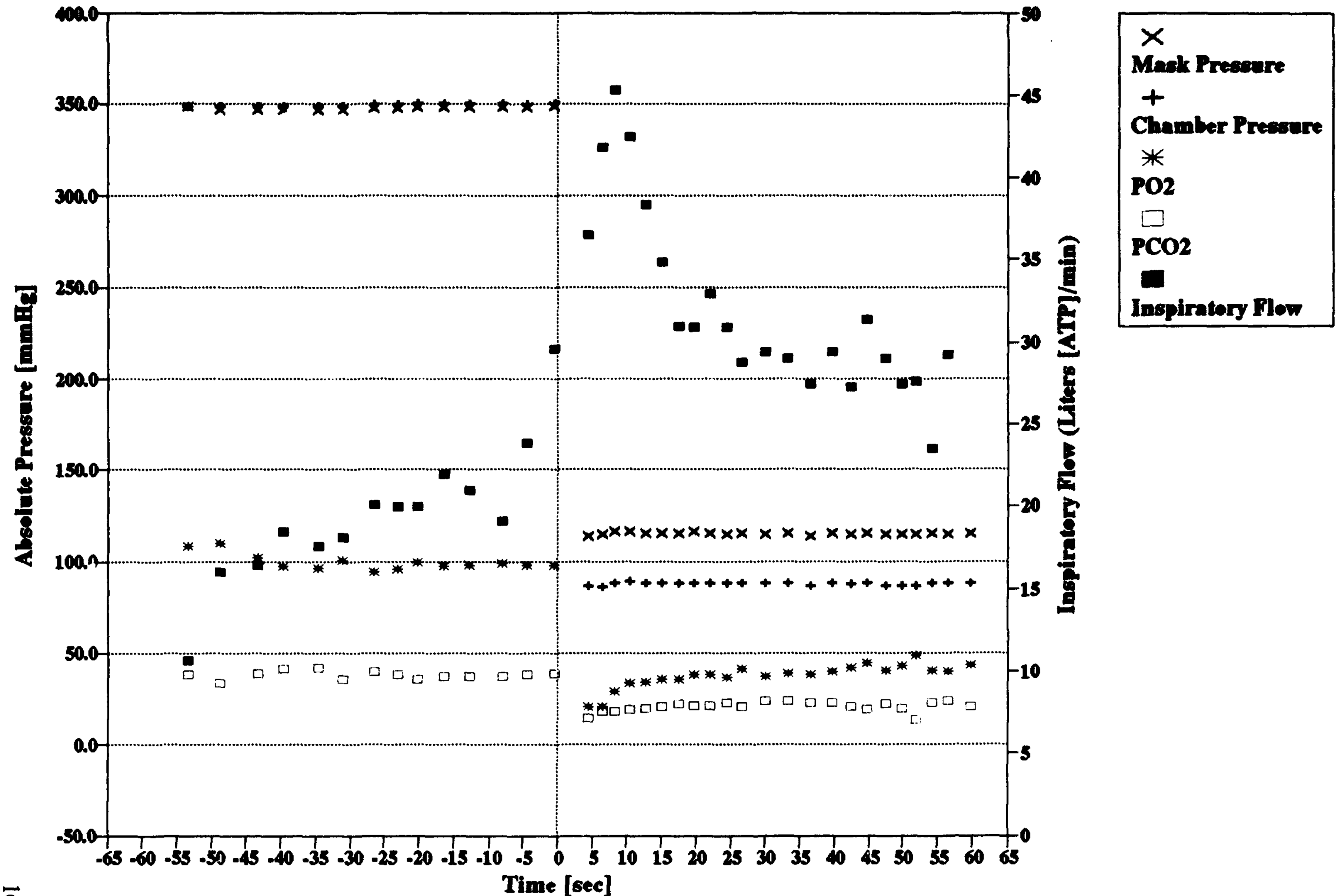
Subject: COX / 100% O2 Dilution
20/50 kft Rapid Decompression

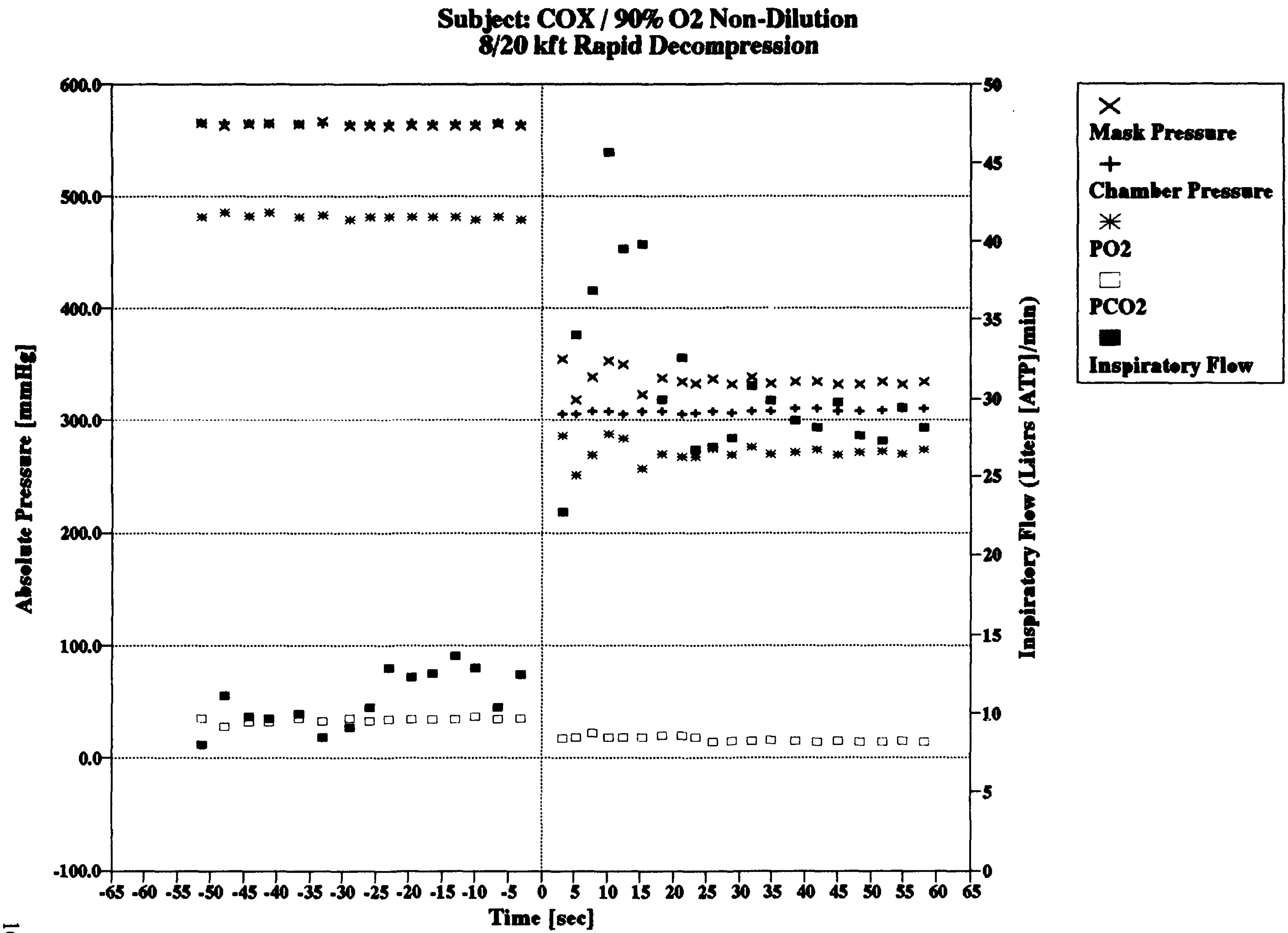


Subject: COX / 93% O2 Non-Dilution
20/50 kft Rapid Decompression

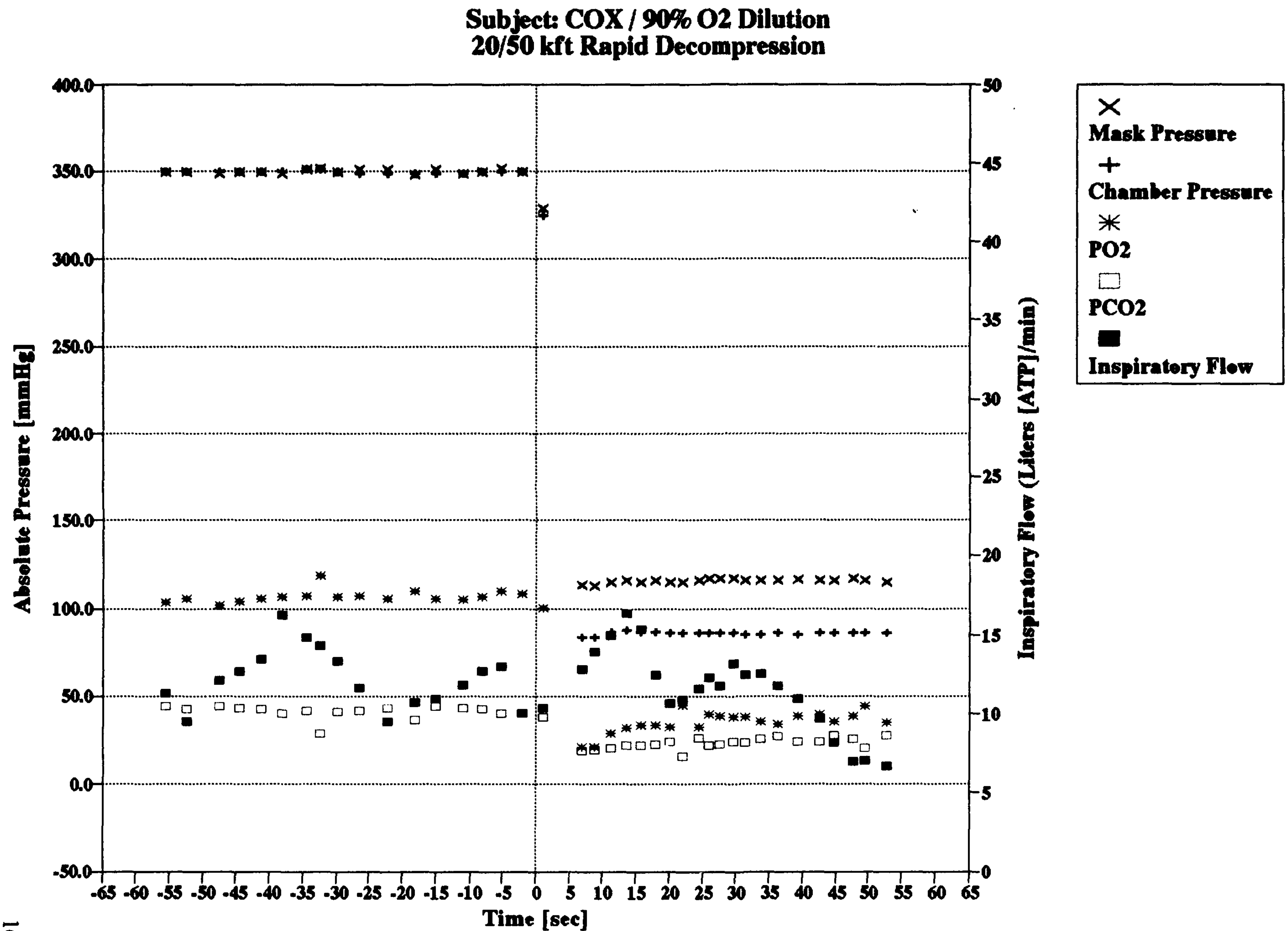


Subject: COX / 93% O2 Dilution
20/50 kft Rapid Decompression

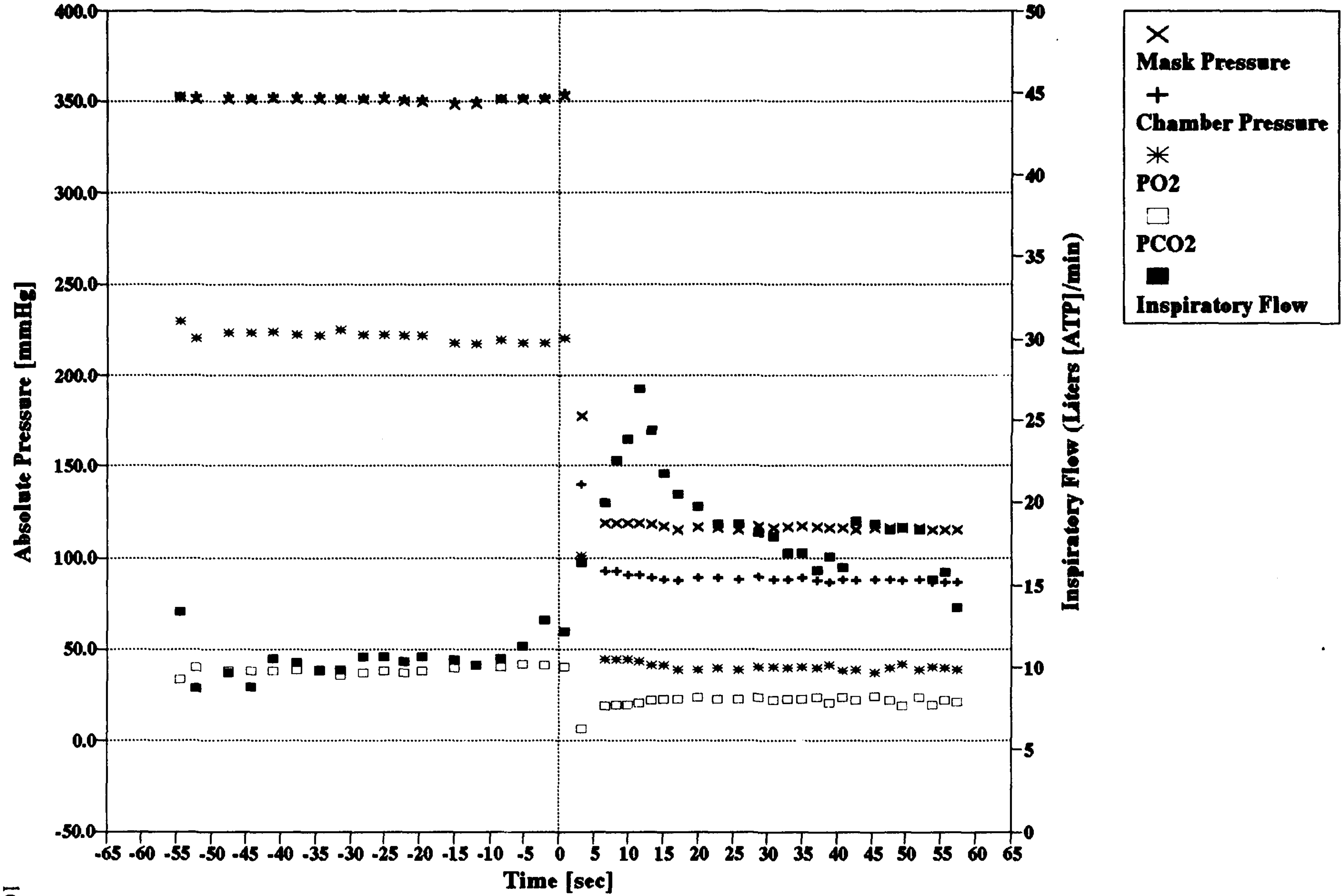




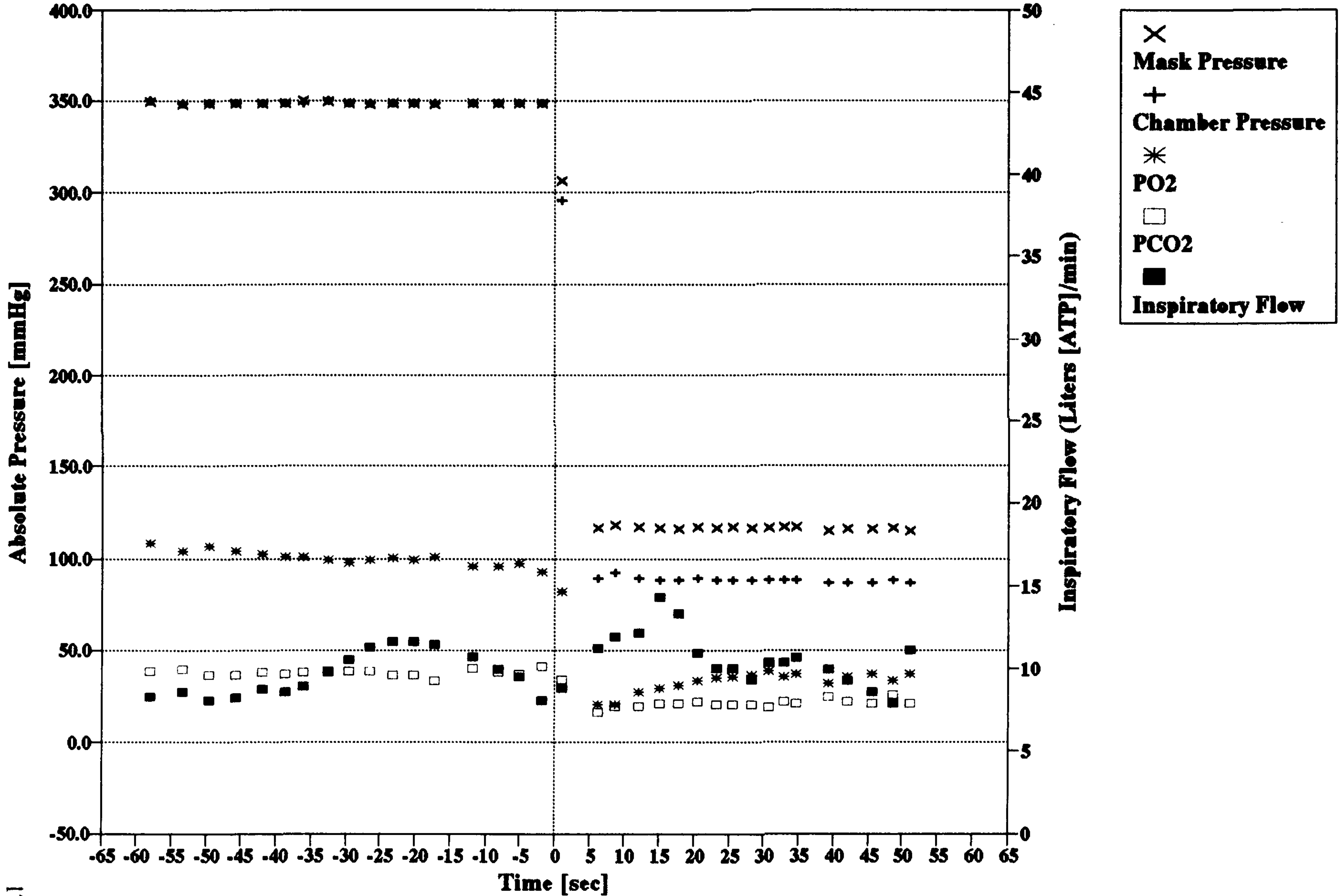
Subject: COX / 90% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow


Subject: COX / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

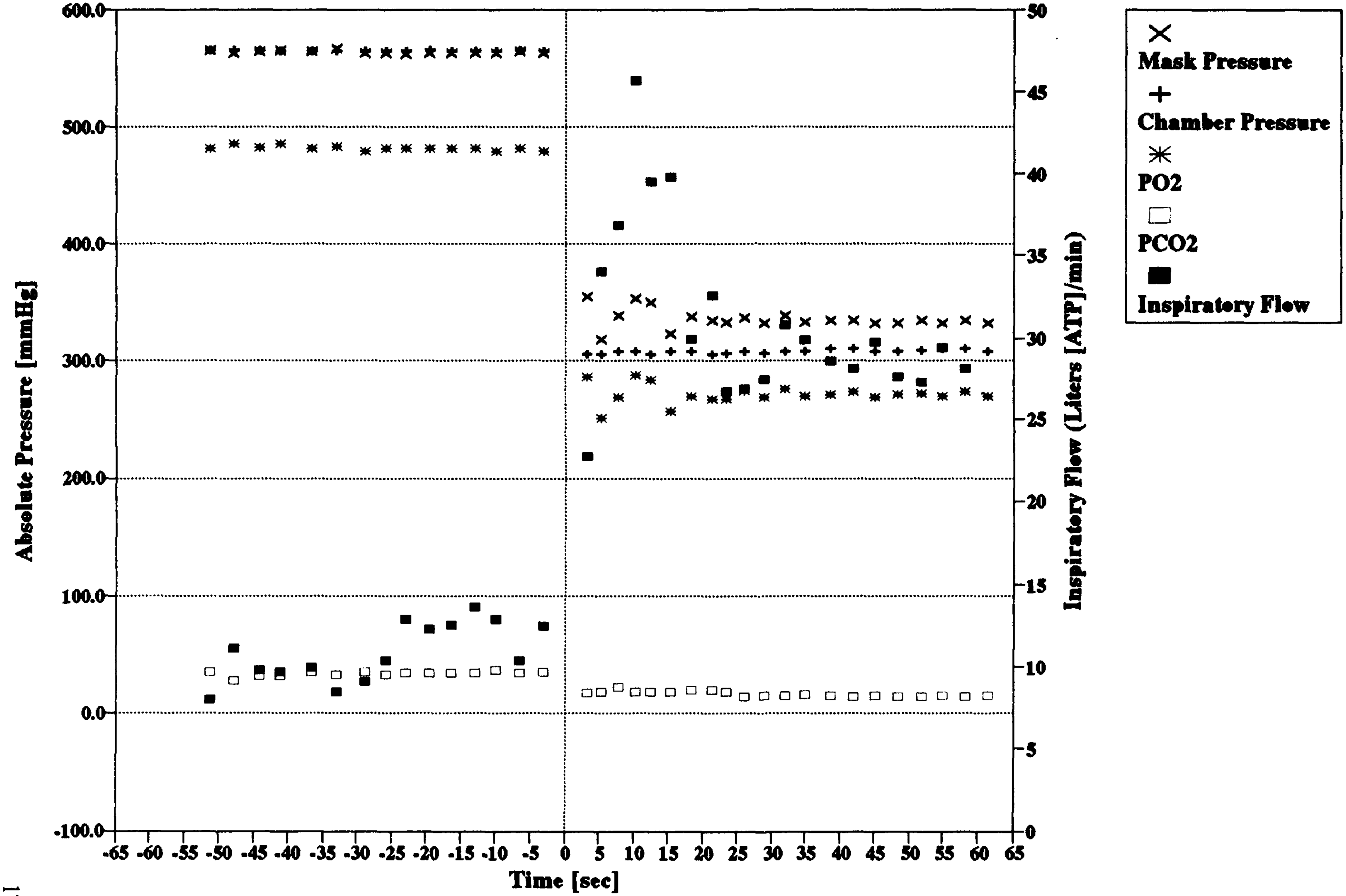
Subject: COX / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



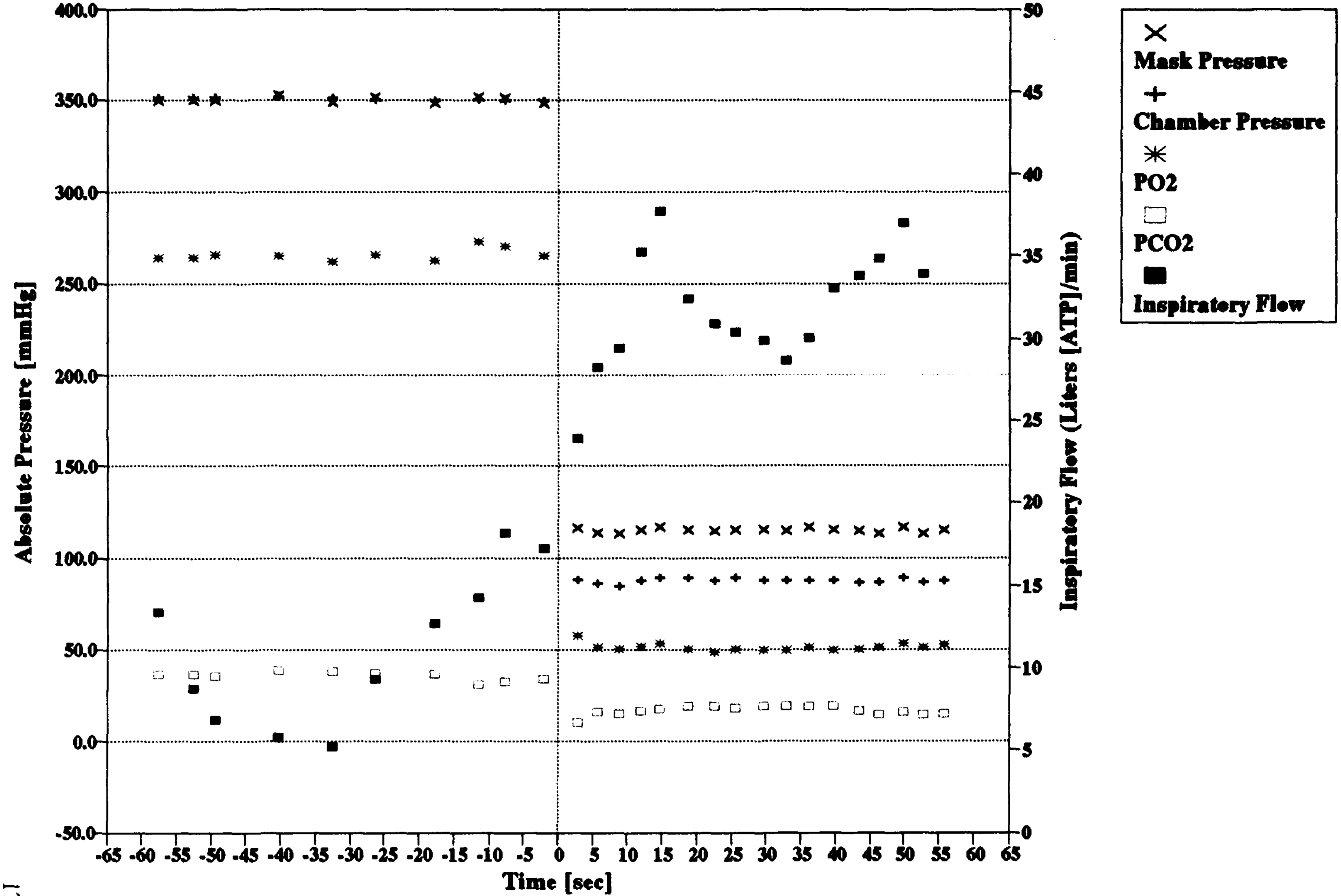
Subject: COX / 85% O2 Dilution
20/50 kft Rapid Decompression



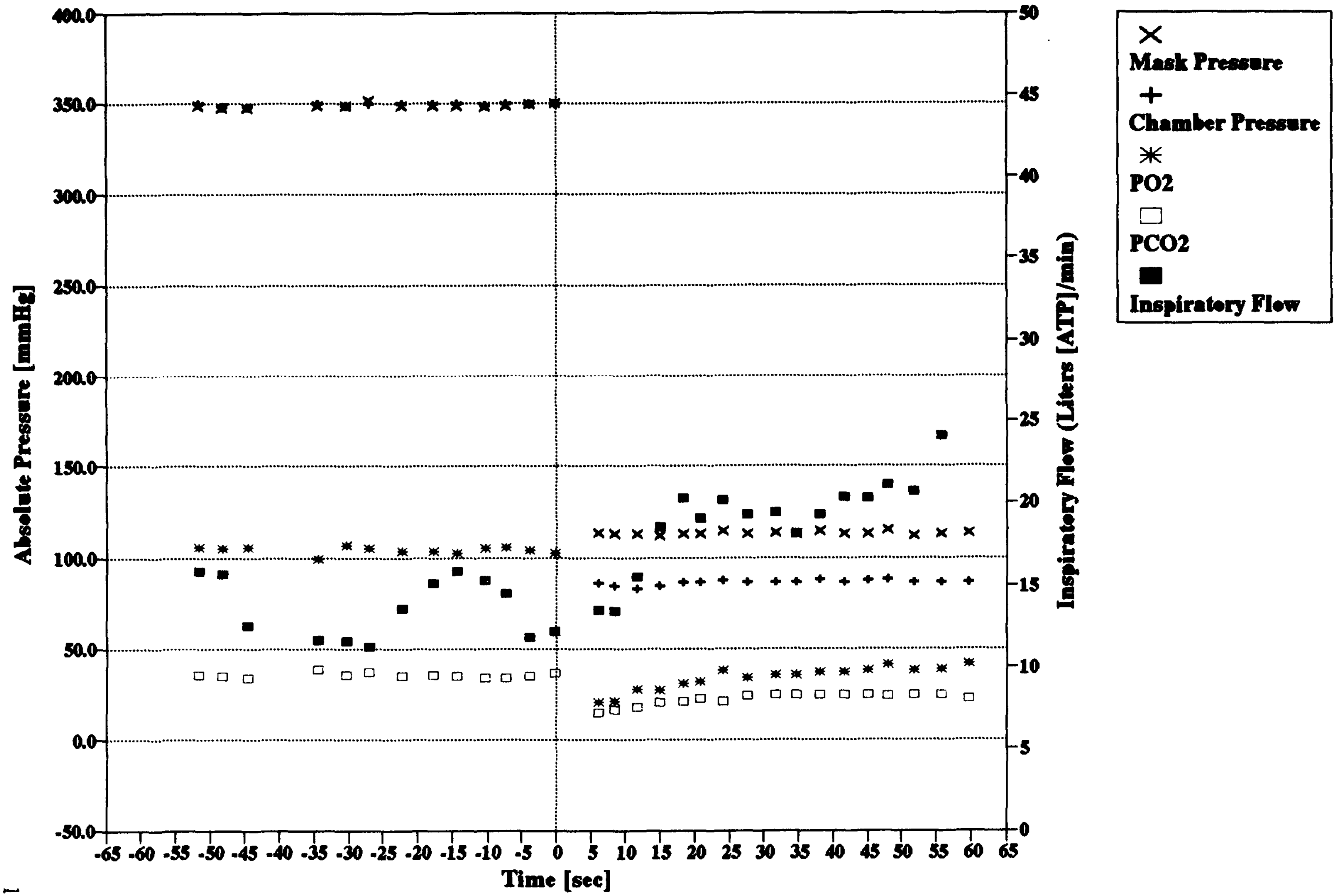
Subject: COX / 100% O2 Non-Dilution
8/20 kft Rapid Decompression

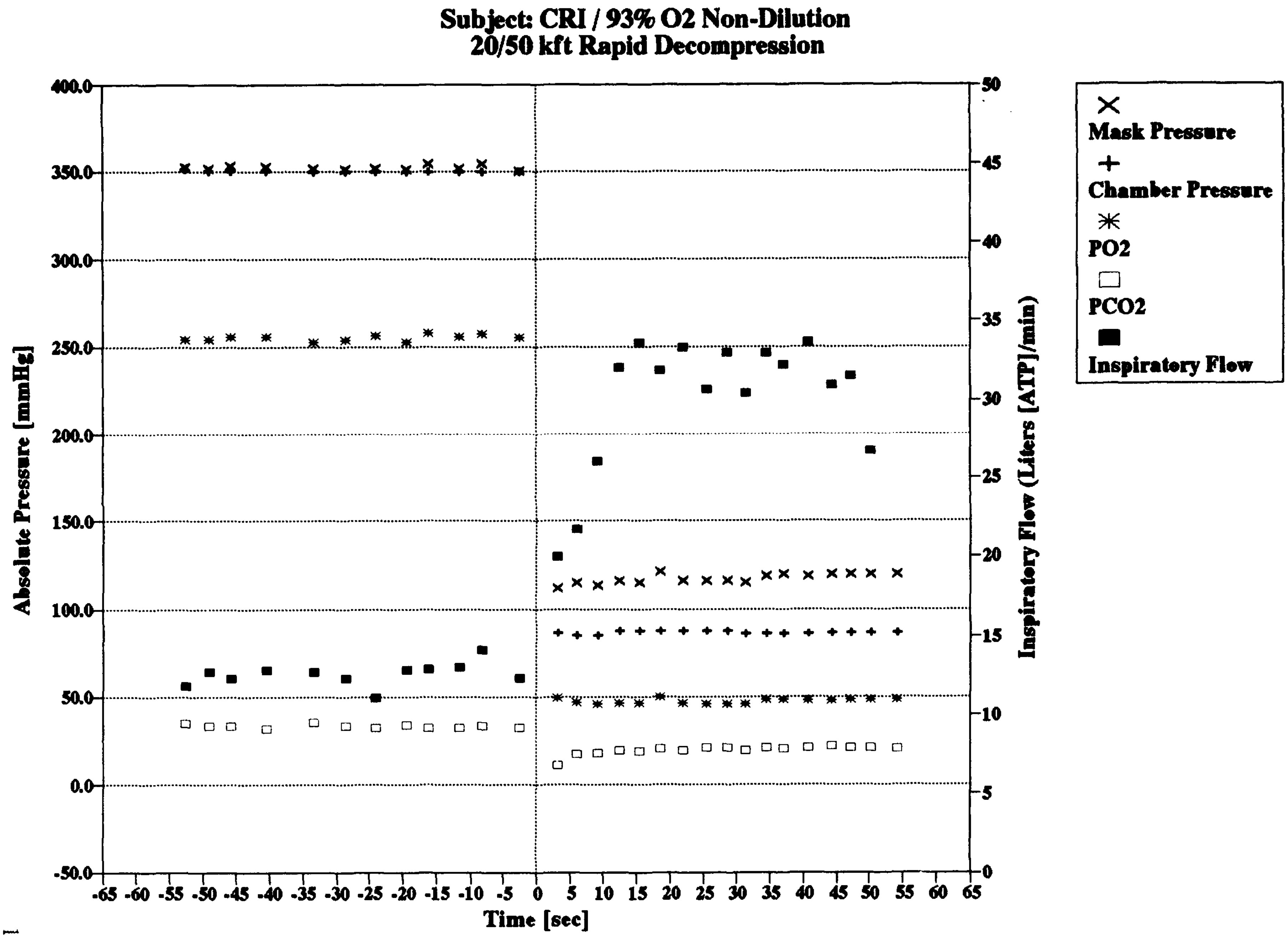


Subject: CRI / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



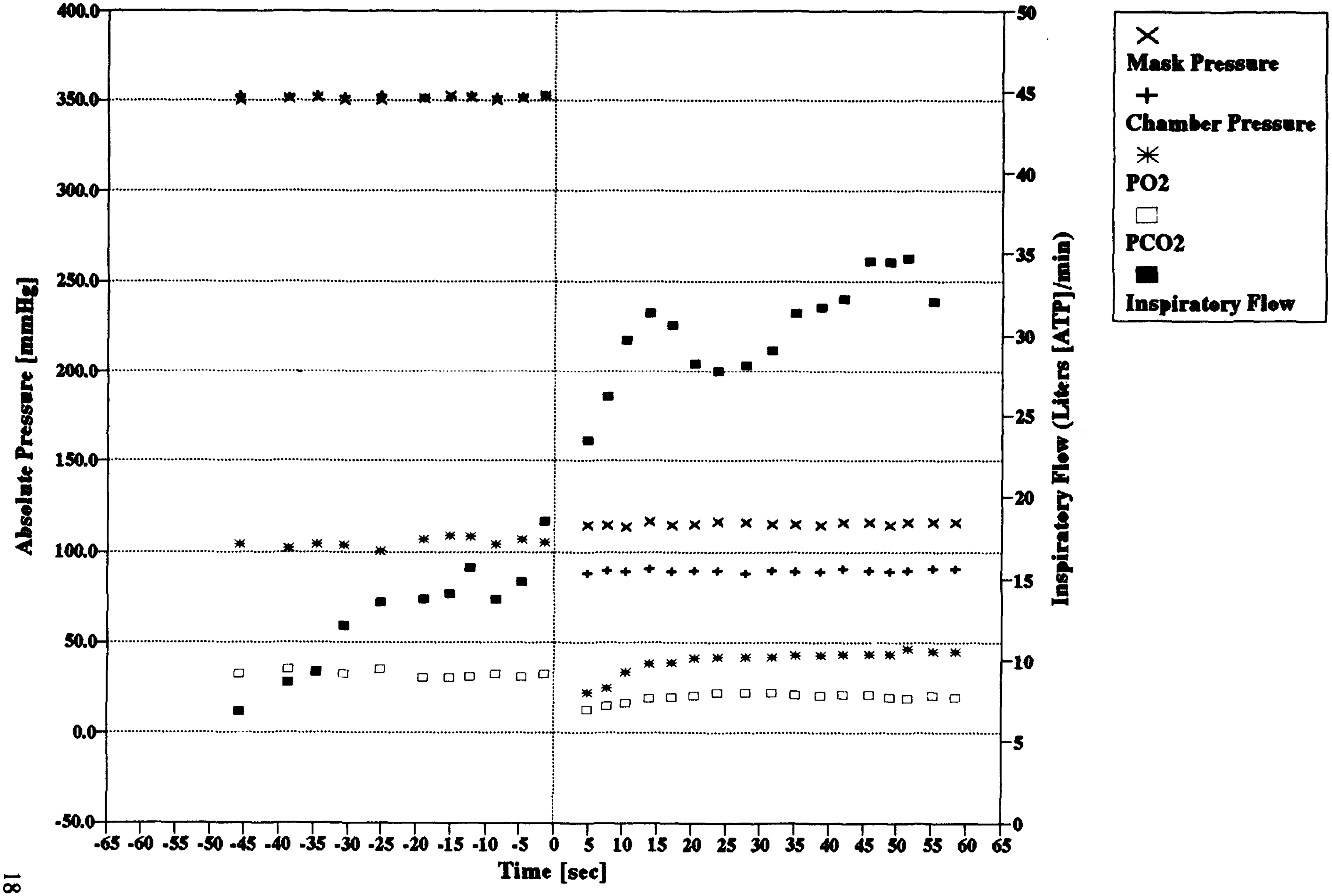
Subject: CRI / 100% O2 Dilution
20/50 kft Rapid Decompression

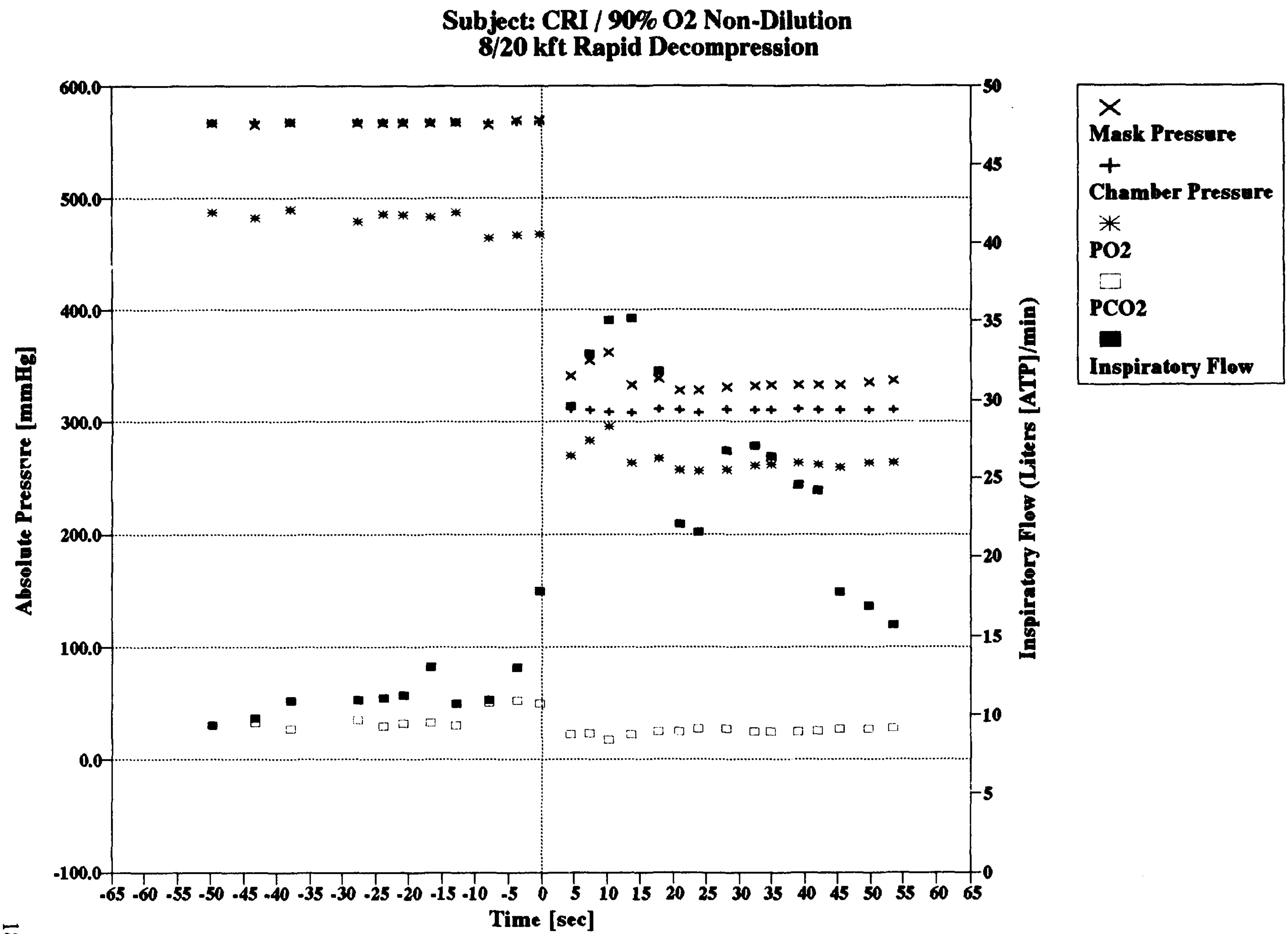




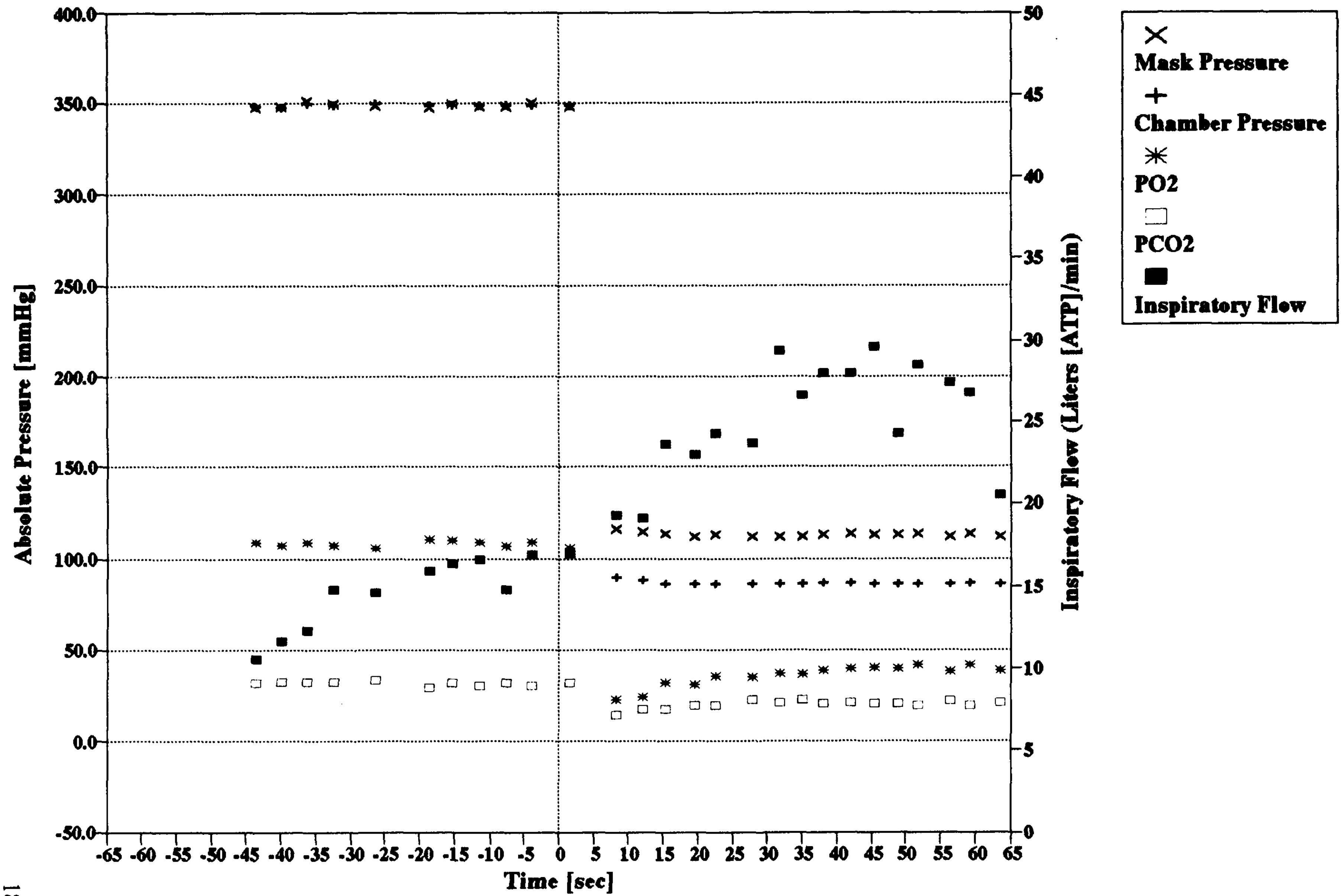
Subject: CRI / 93% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

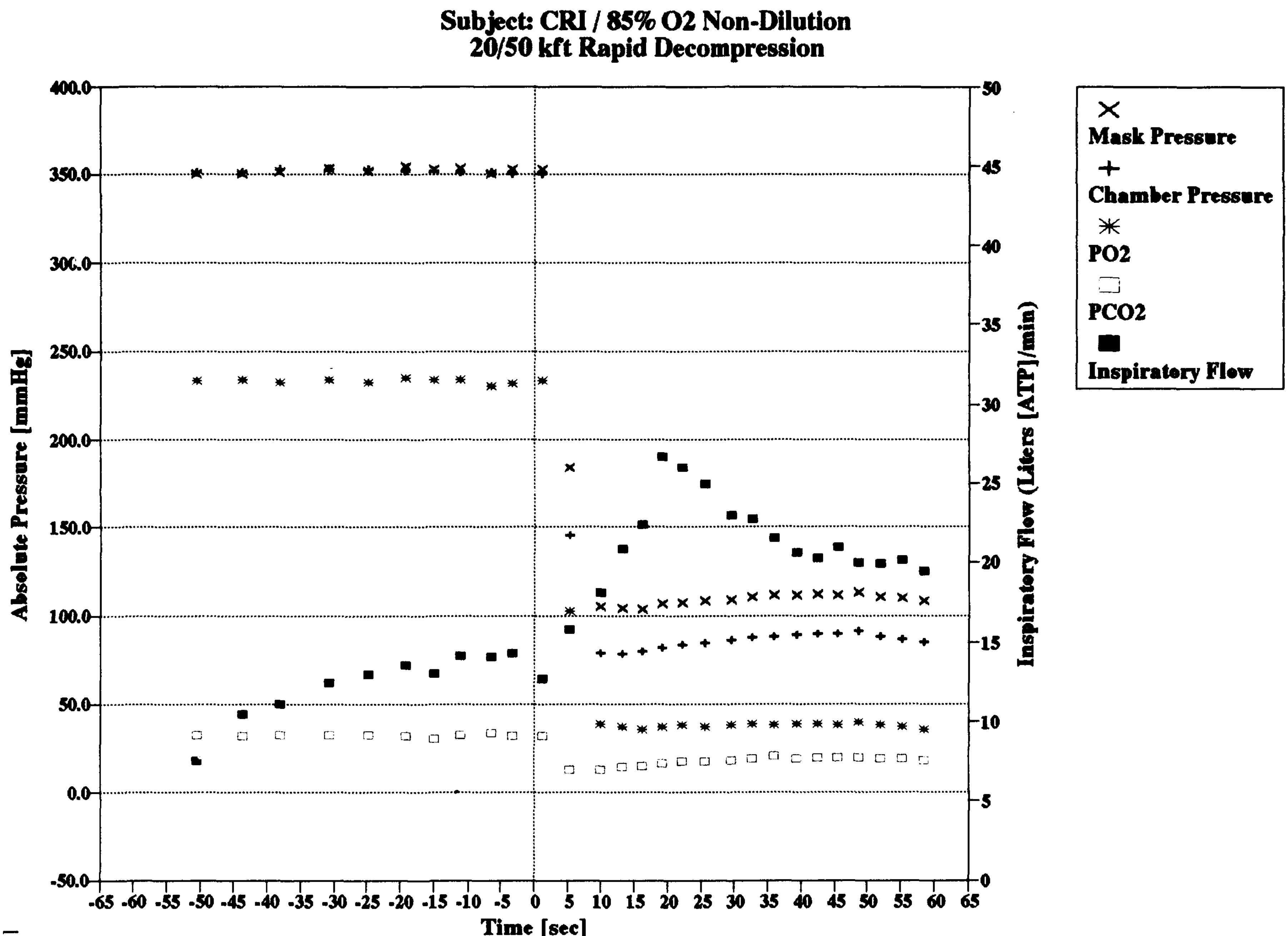
Subject: CRI / 93% O2 Dilution
20/50 kft Rapid Decompression




Subject: CRI / 90% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

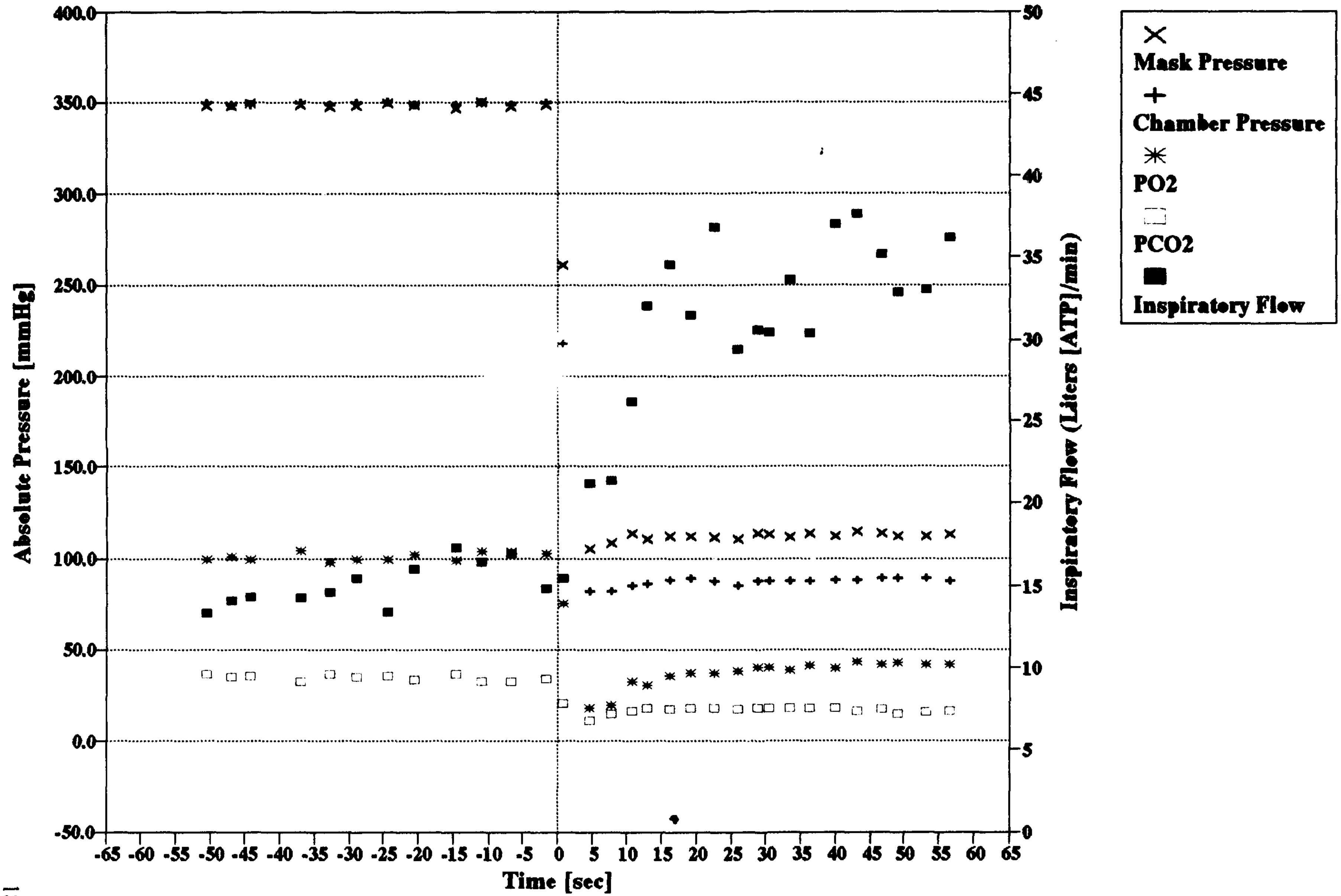
Subject: CRI / 90% O2 Dilution
20/50 kft Rapid Decompression



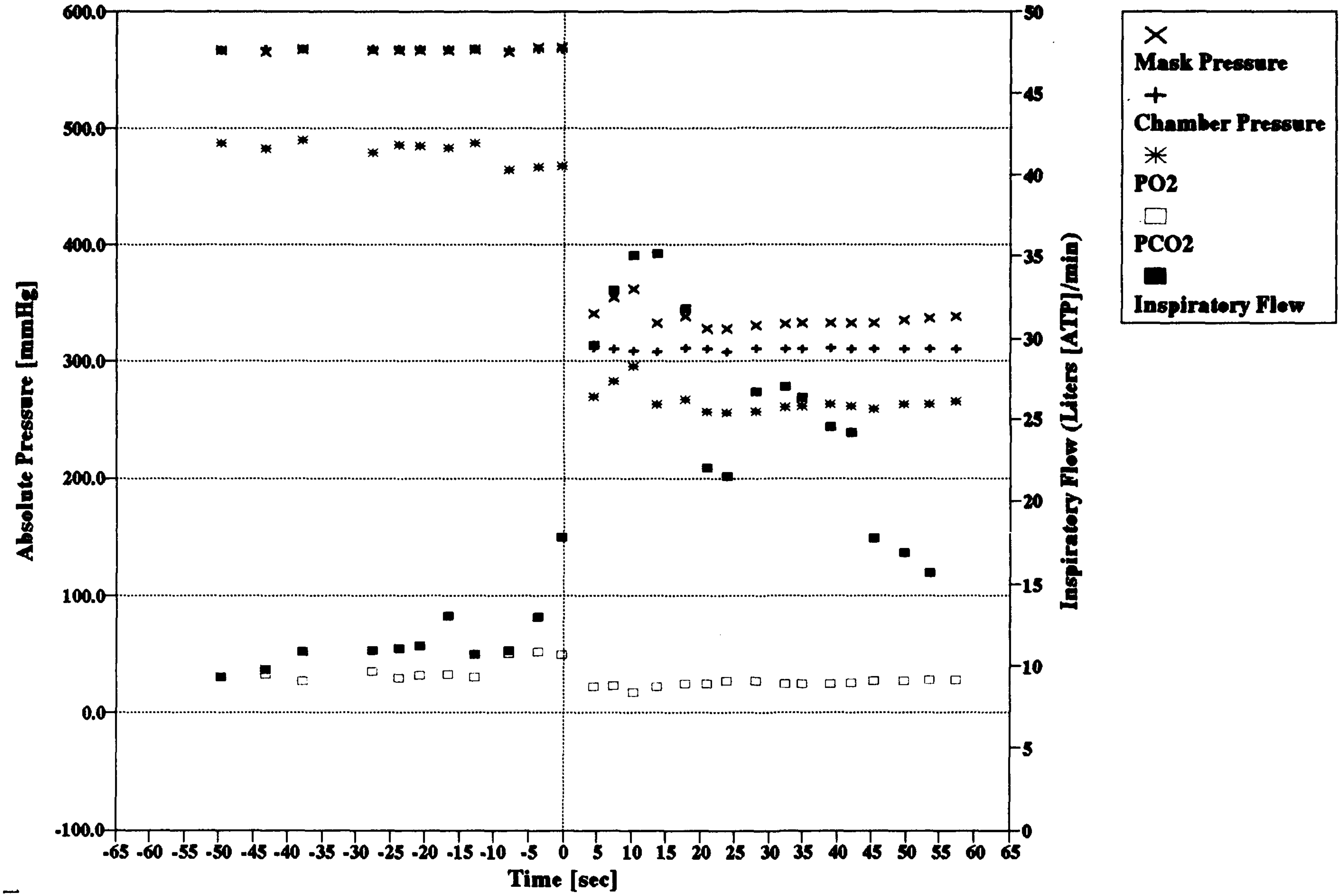


Subject: CRI / 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

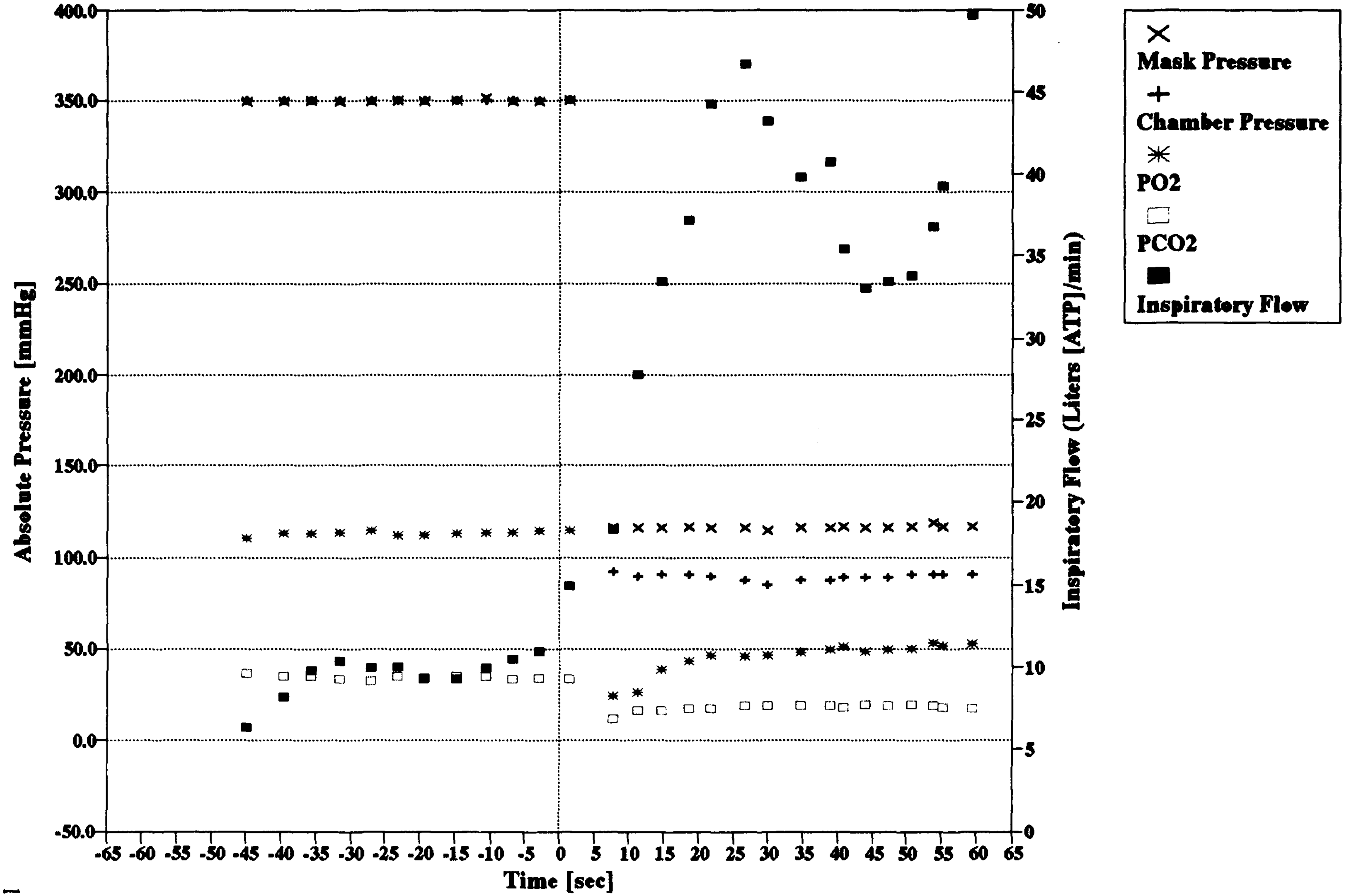
Subject: CRI / 85% O2 Dilution
20/50 kft Rapid Decompression



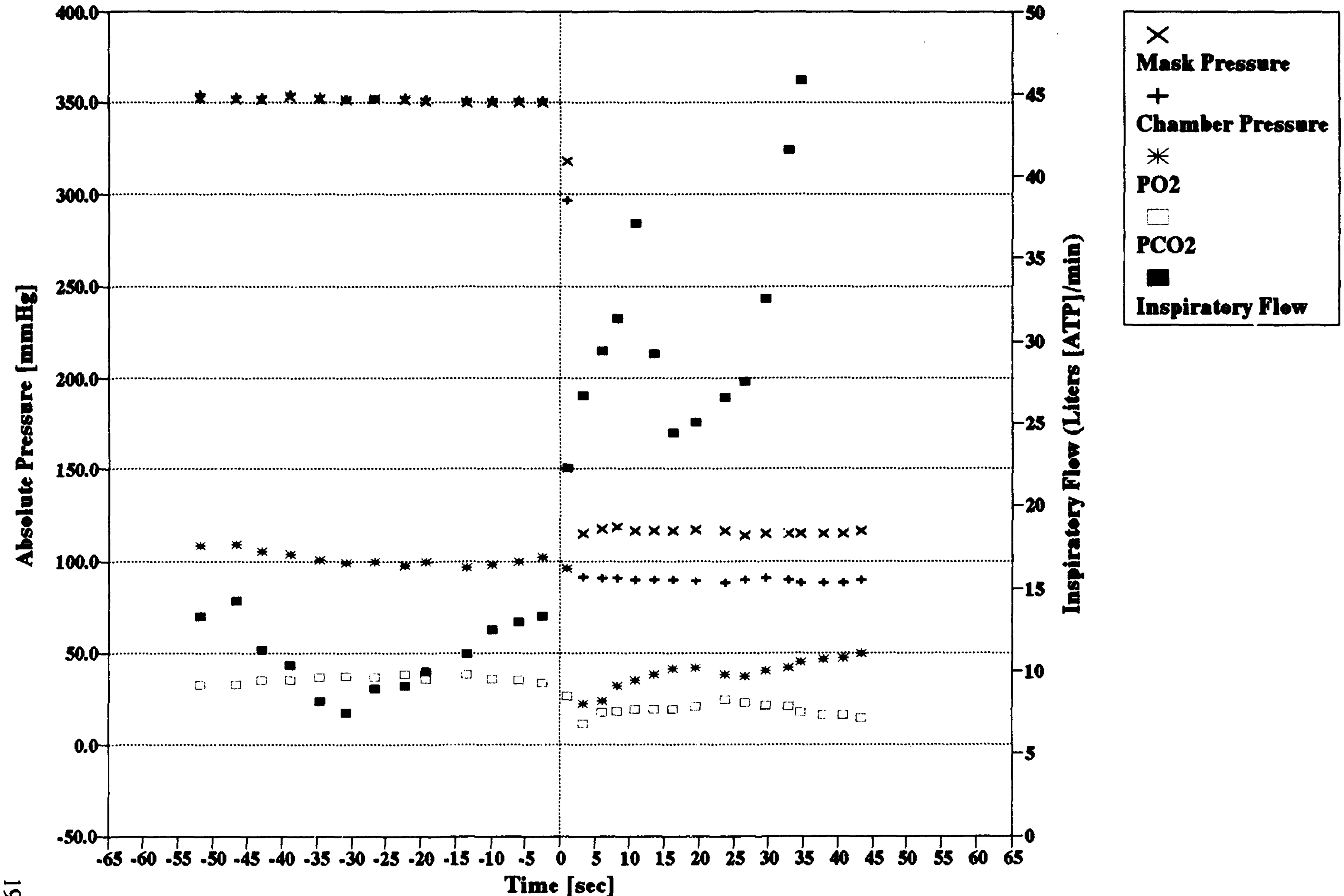
Subject: CRI / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



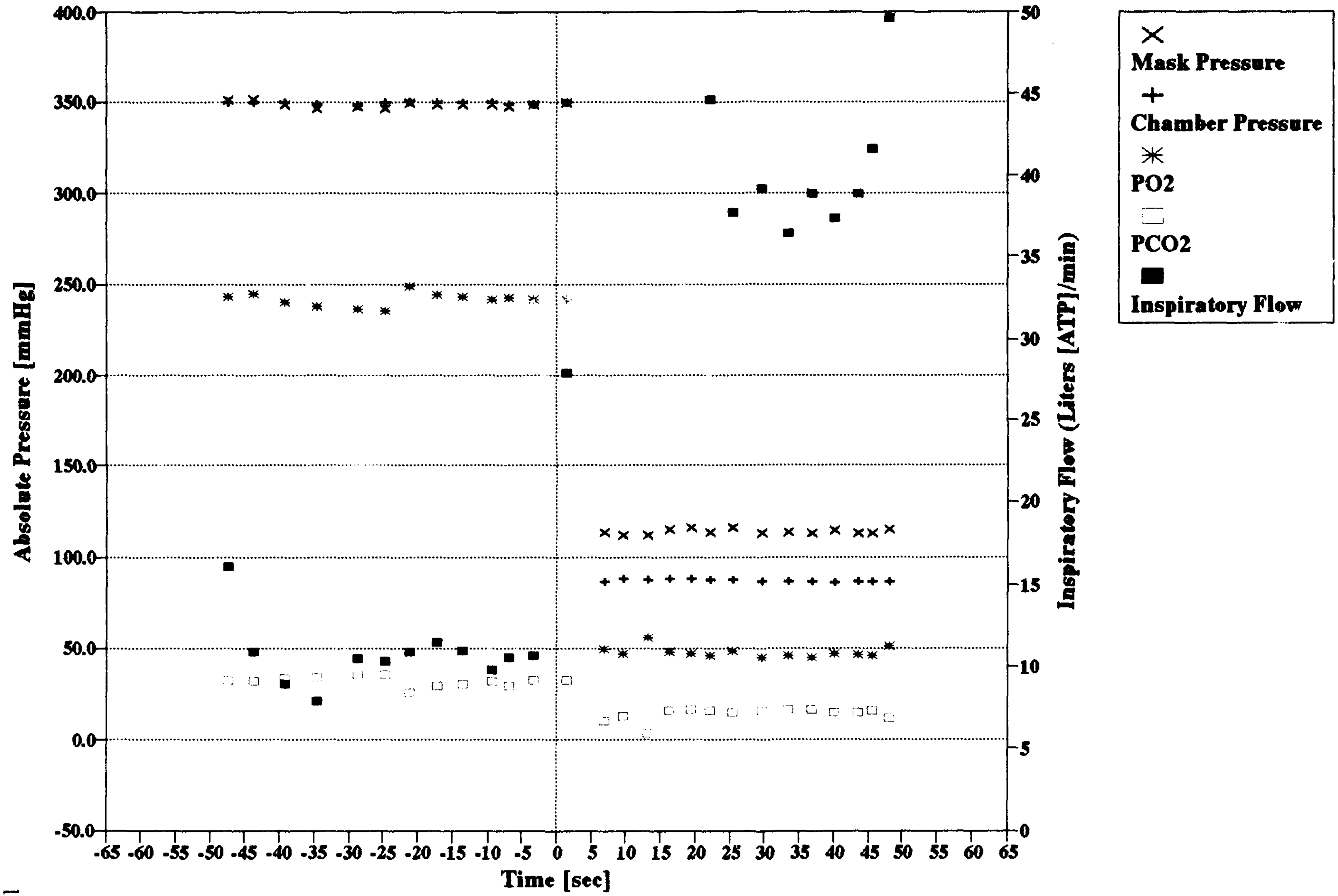
Subject: DOA / 100% O2 Dilution
20/50 kft Rapid Decompression

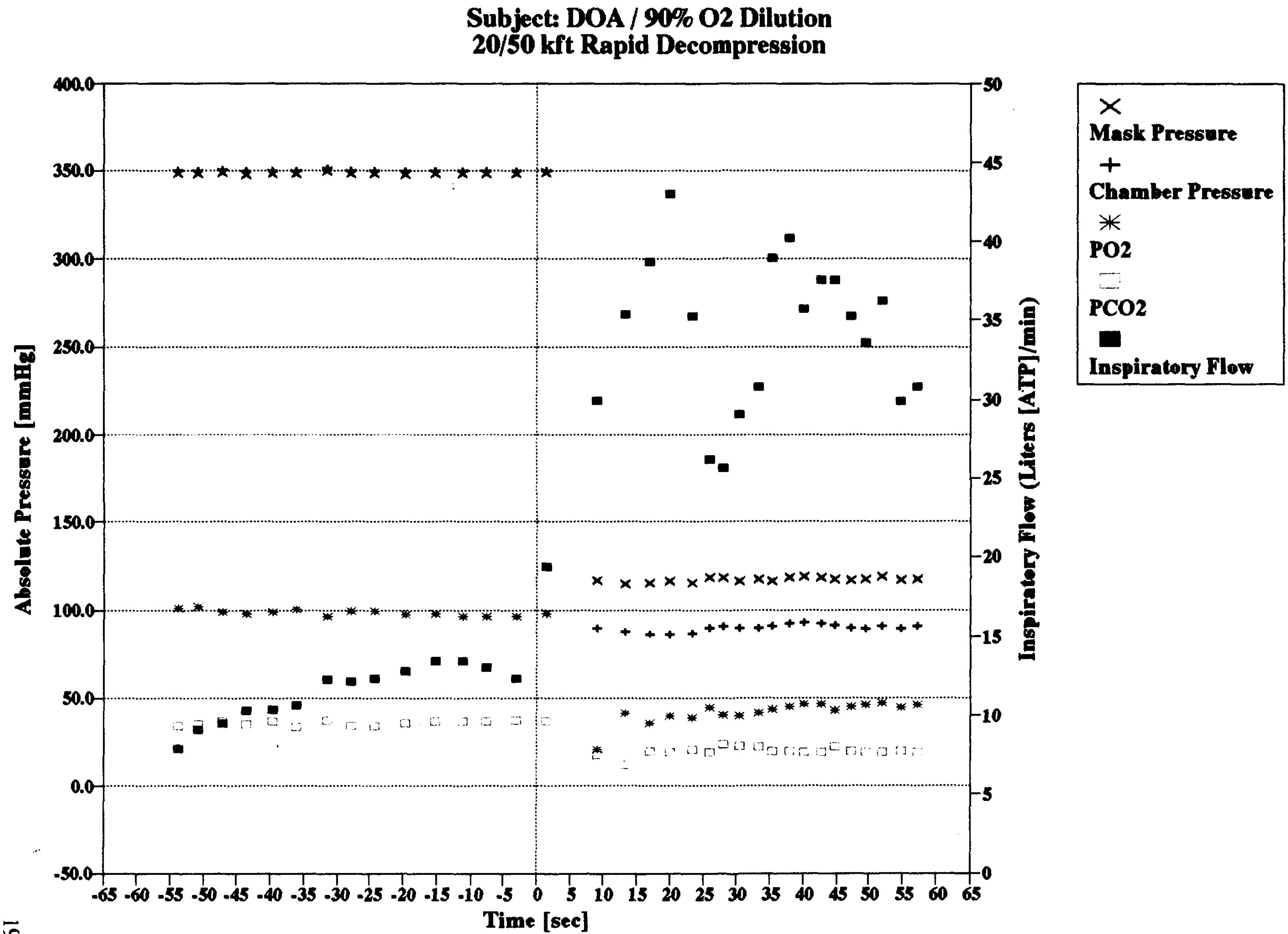


Subject: DOA / 93% O2 Dilution
20/50 kft Rapid Decompression



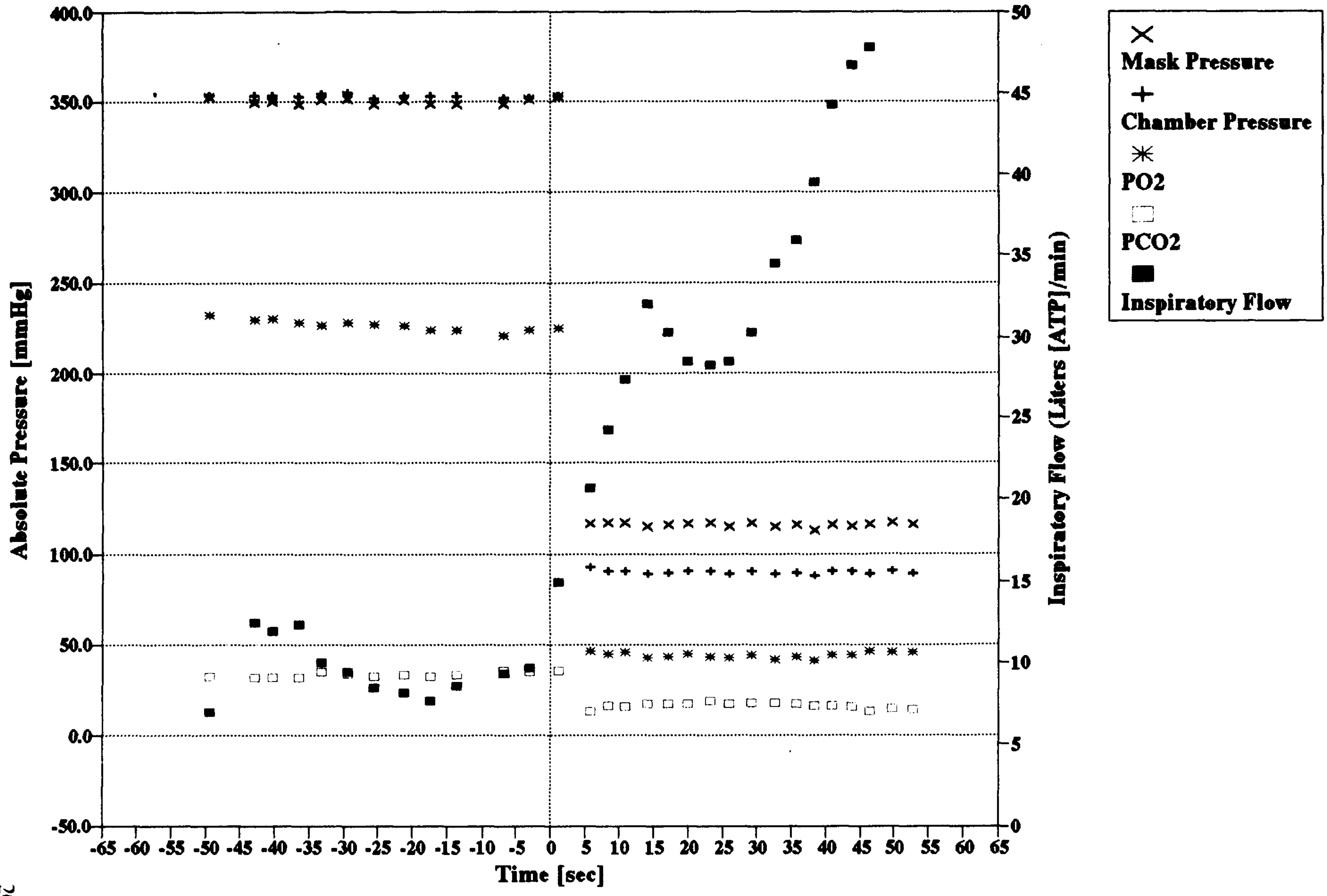
Subject: DOA / 90% O2 Non-Dilution
20/50 kft Rapid Decompression



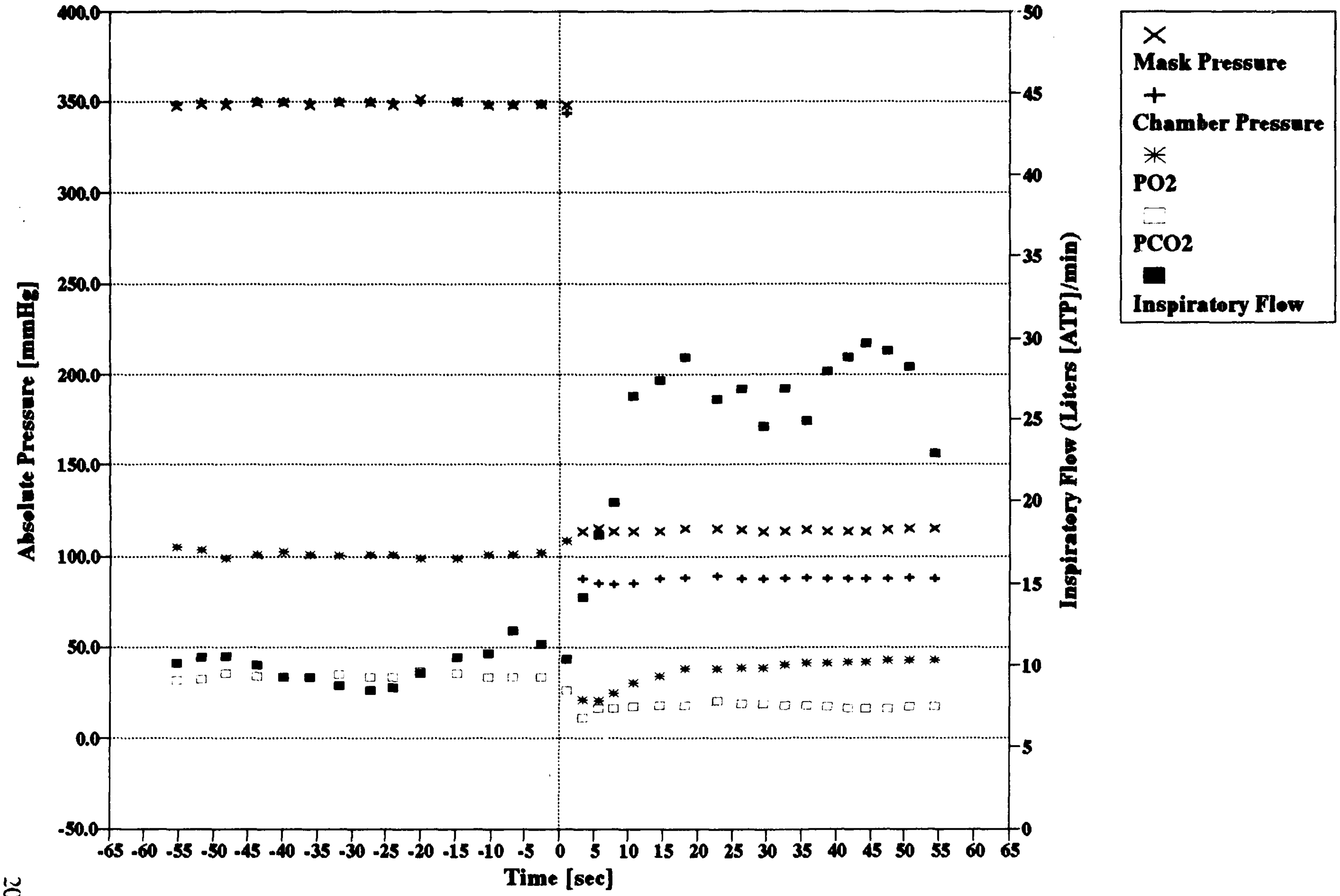


Subject: DOA / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

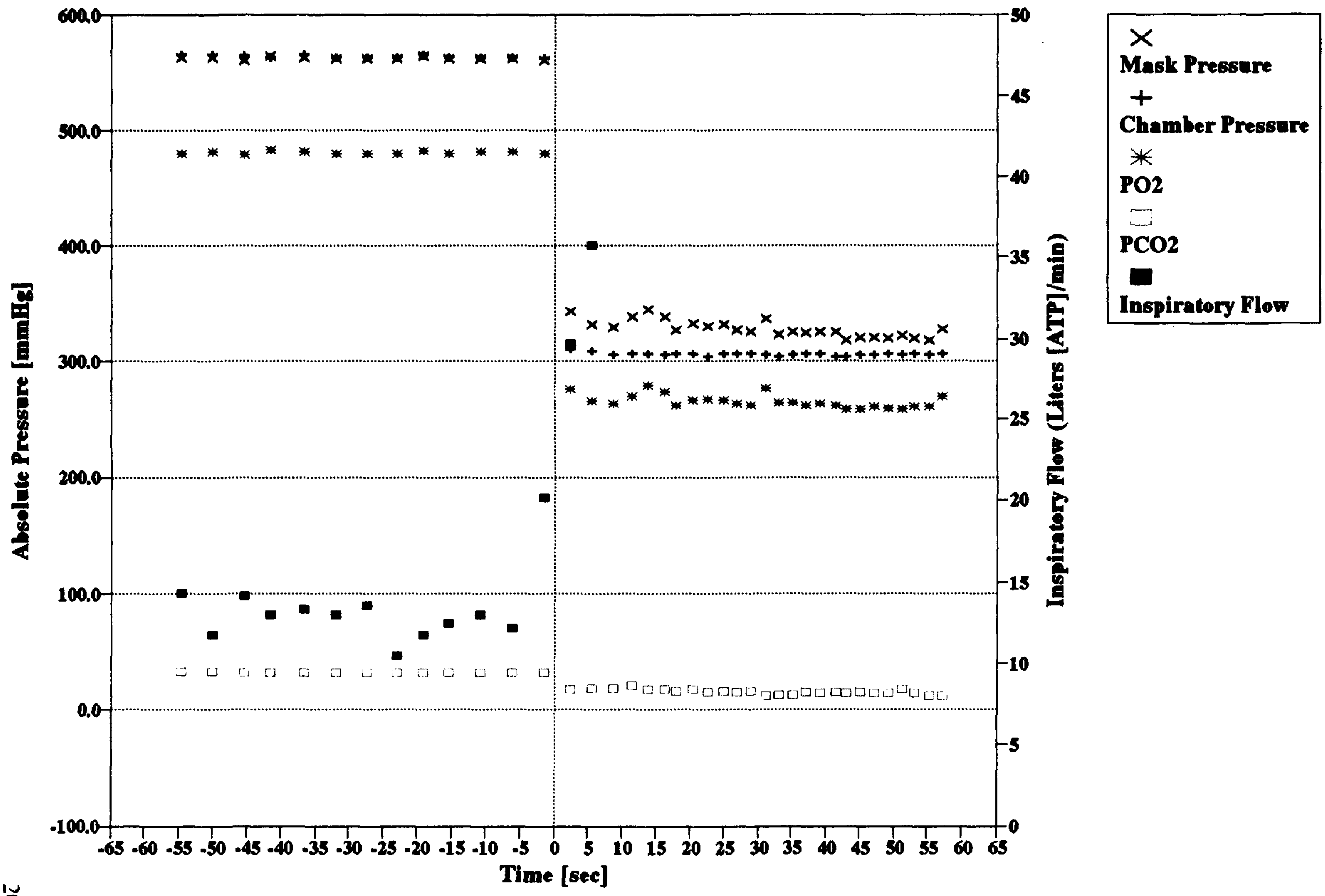
Subject: DOA / 85% O2 Non-Dilution
20/50 kft Rapid Decompression

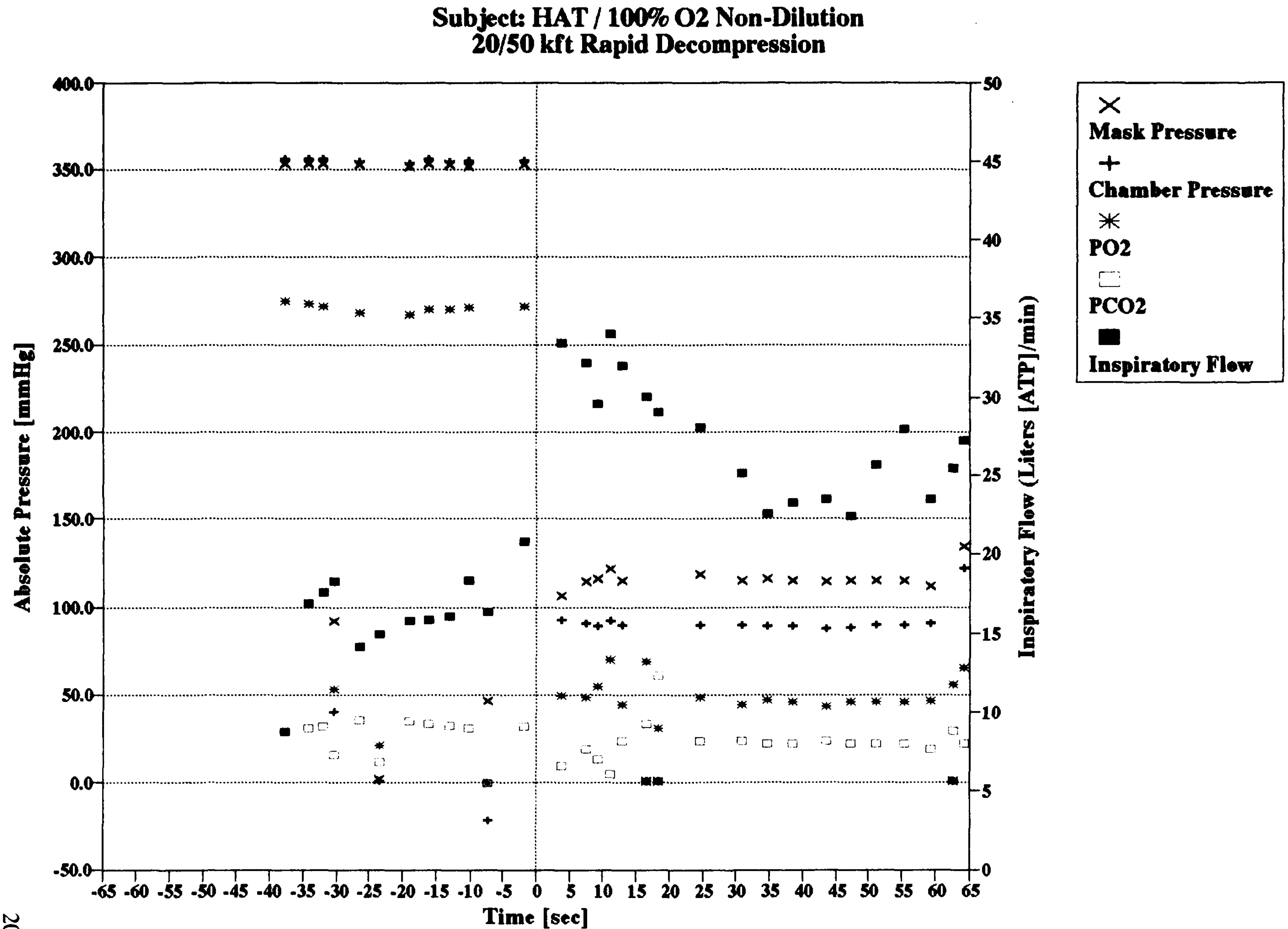


Subject: DOA / 85% O2 Dilution
20/50 kft Rapid Decompression

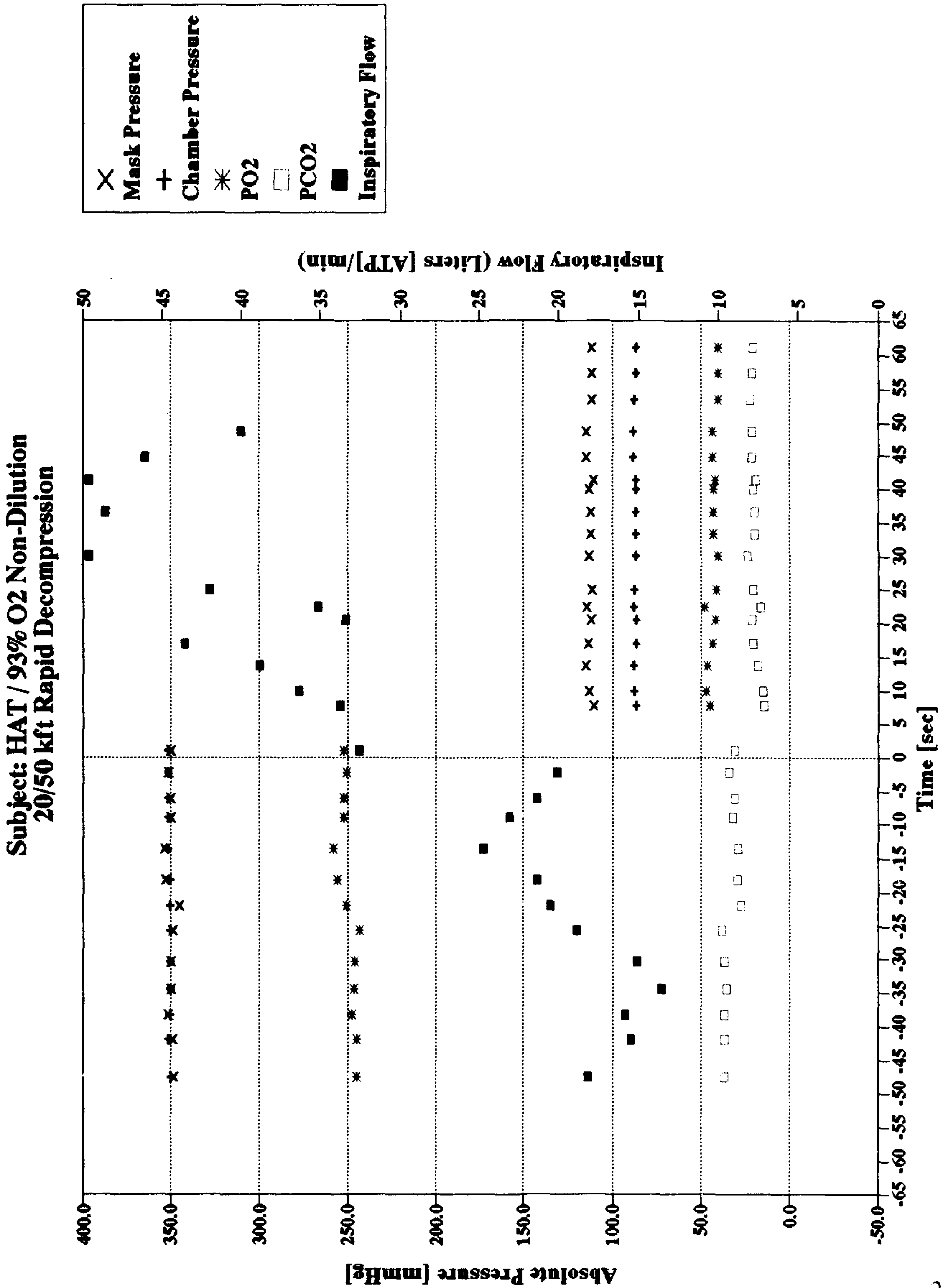


Subject: DOA / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



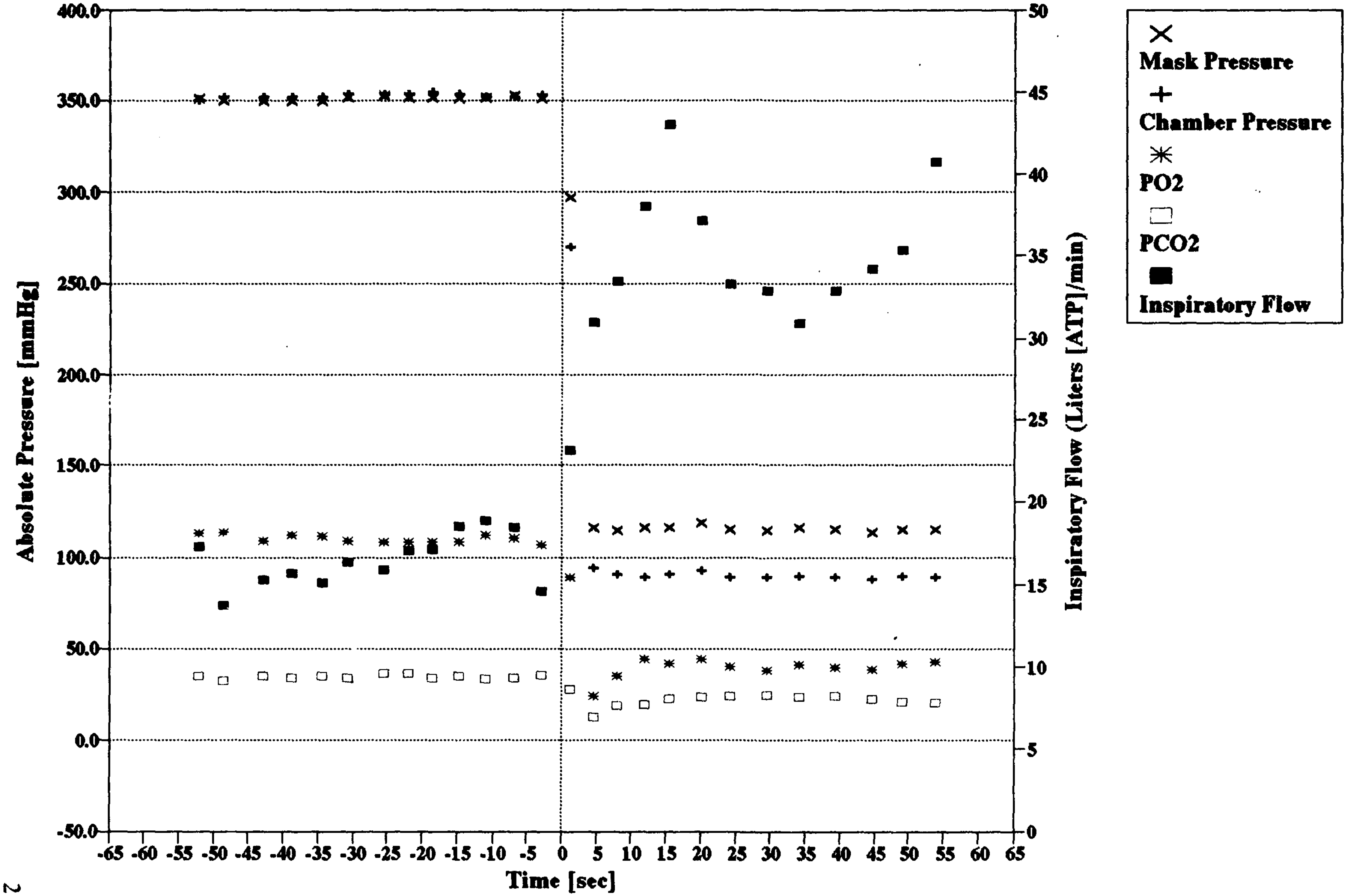


Subject: HAT / 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

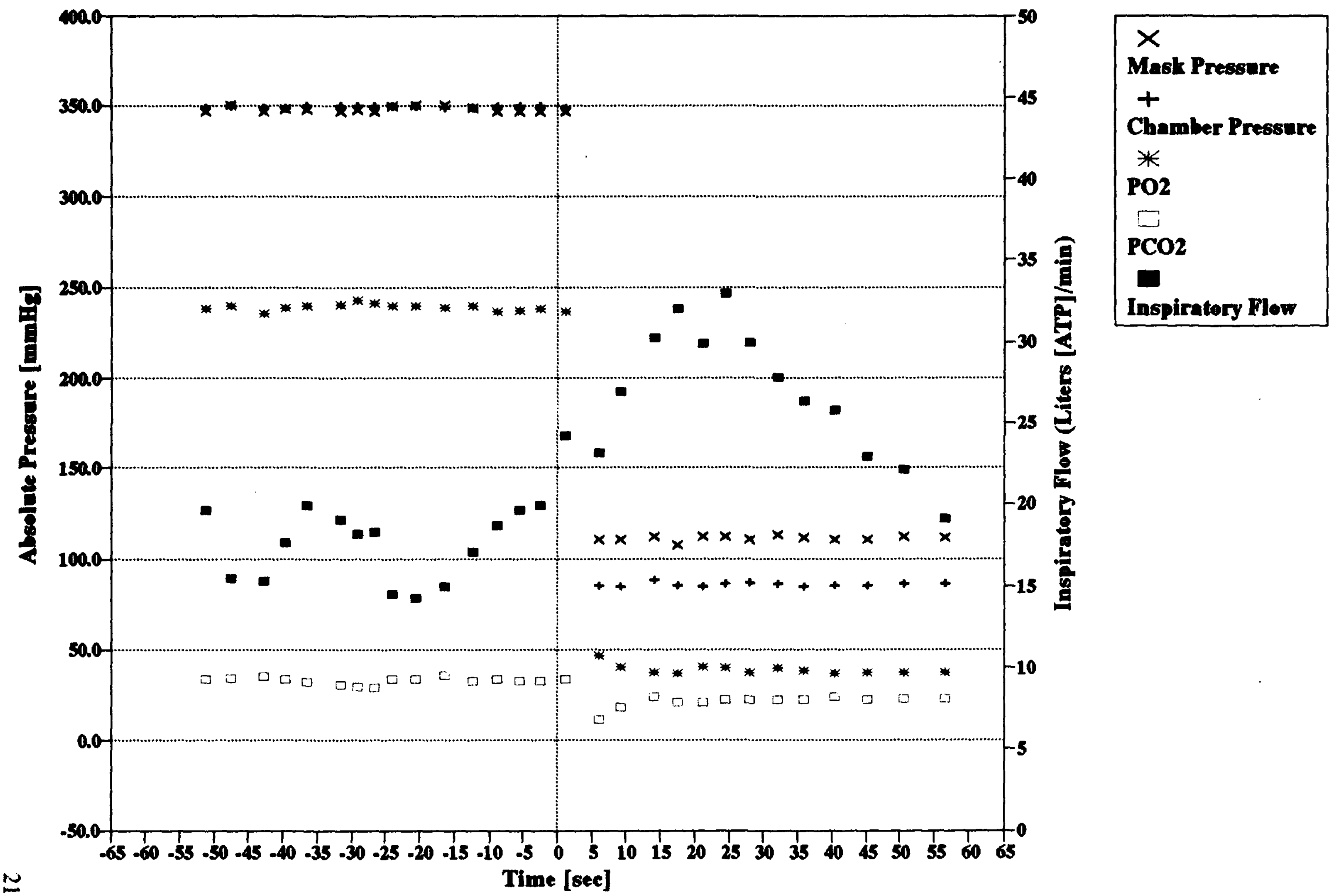


Subject: HAT / 93% O2 Non-Dilution
20/50 kft Rapid Decompression
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow
Inspiratory Flow (Liters [ATP]/min)
Absolute Pressure [mmHg]
Time [sec]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
50
45
40
35
30
25
20
15
10
5
0
-65
-60
-55
-50
-45
-40
-35
-30
-25
-20
-15
-10
-5
0
5
10
15
20
25
30
35
40
45
50
55
60
65

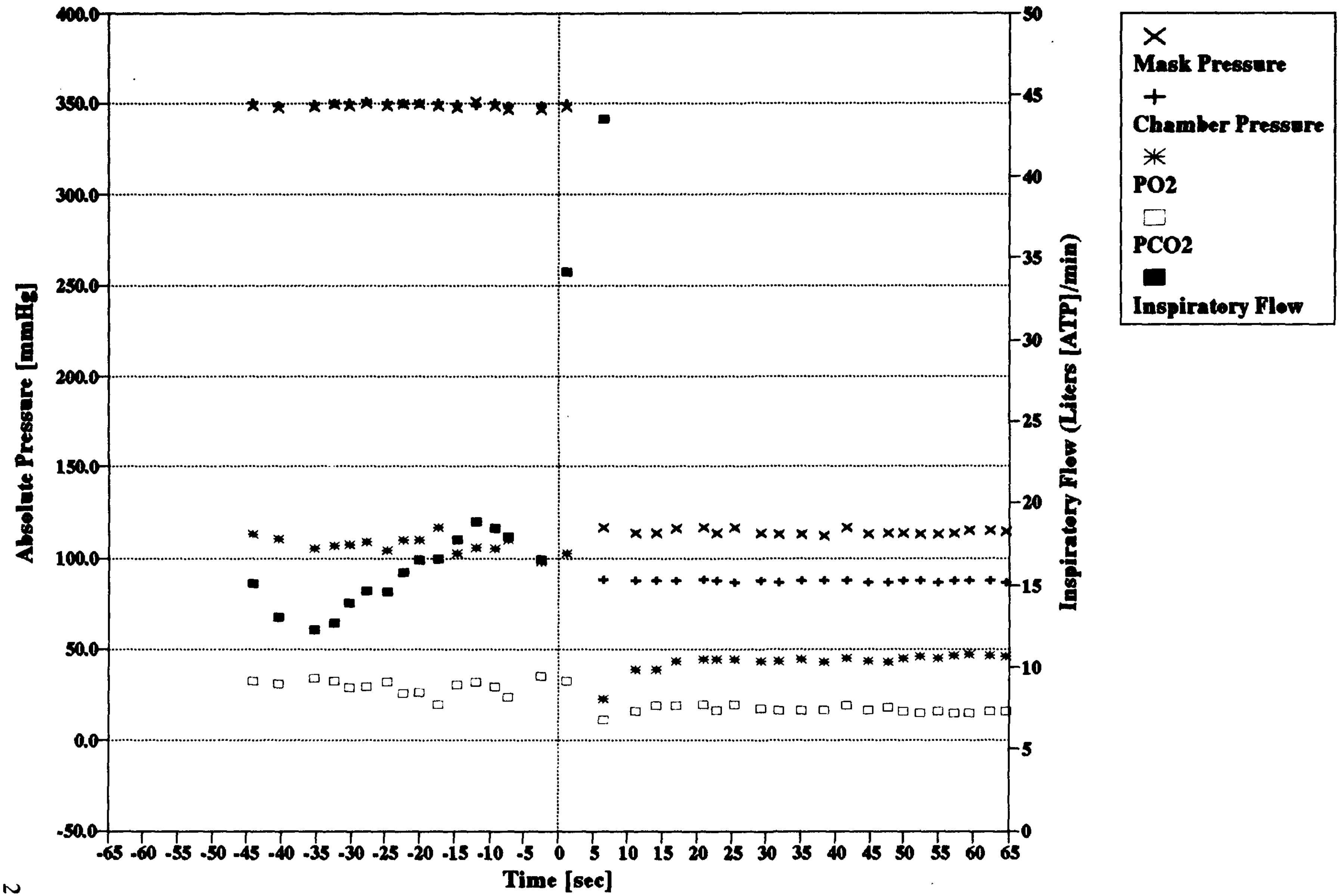
Subject: HAT / 93% O2 Dilution
20/50 kft Rapid Decompression



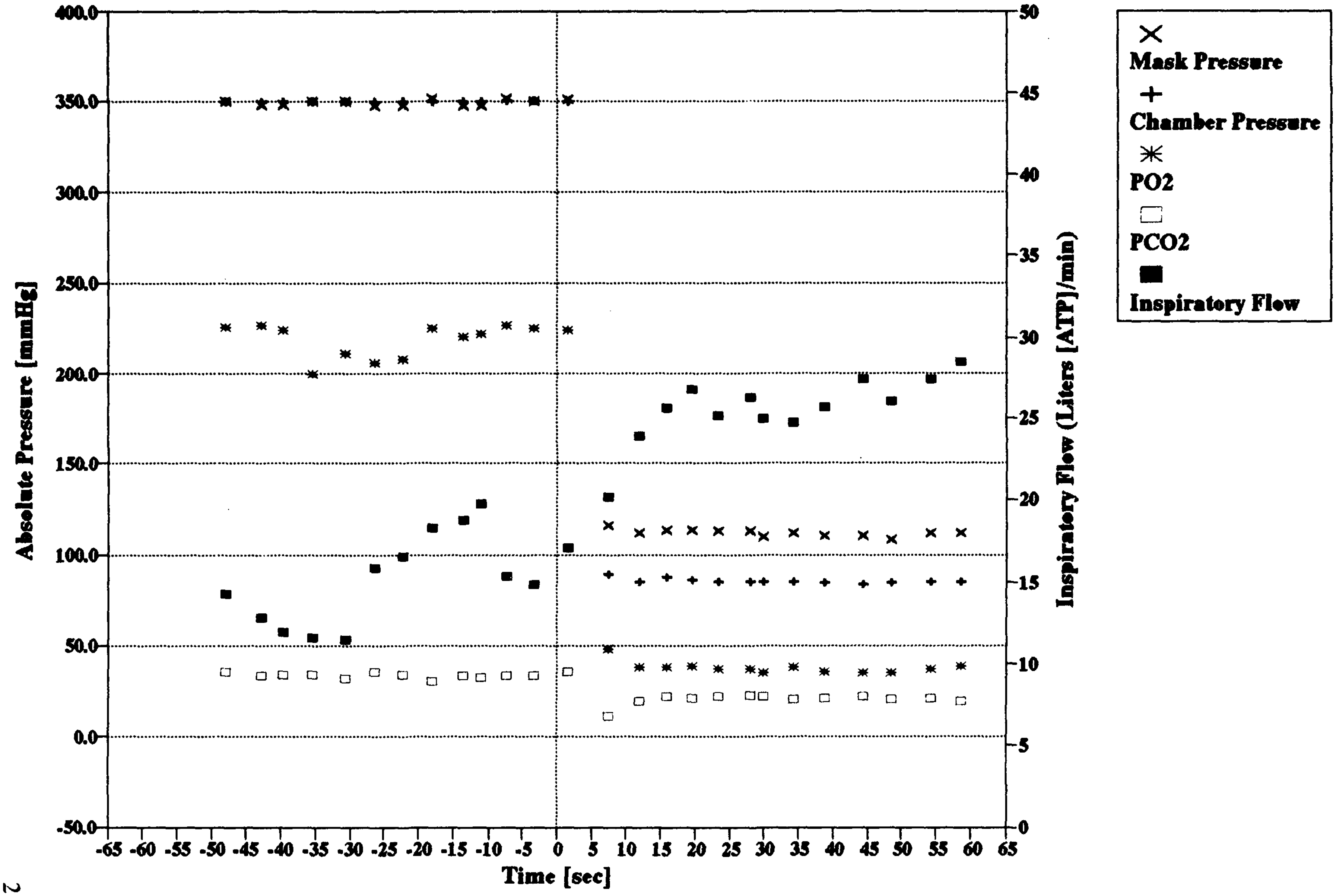
Subject: HAT / 90% O2 Non-Dilution
20/50 kft Rapid Decompression

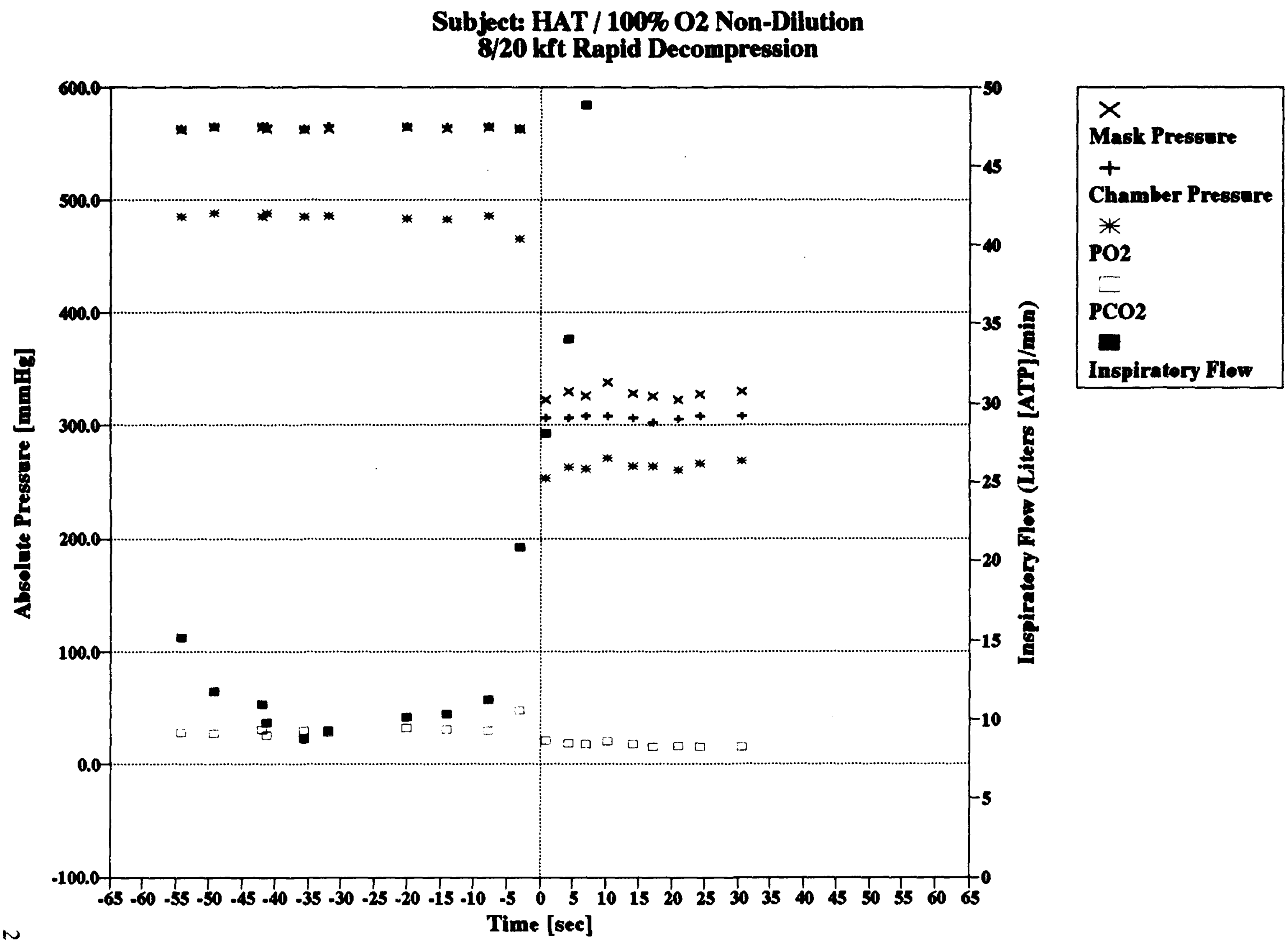


Subject: HAT / 90% O2 Dilution
20/50 kft Rapid Decompression

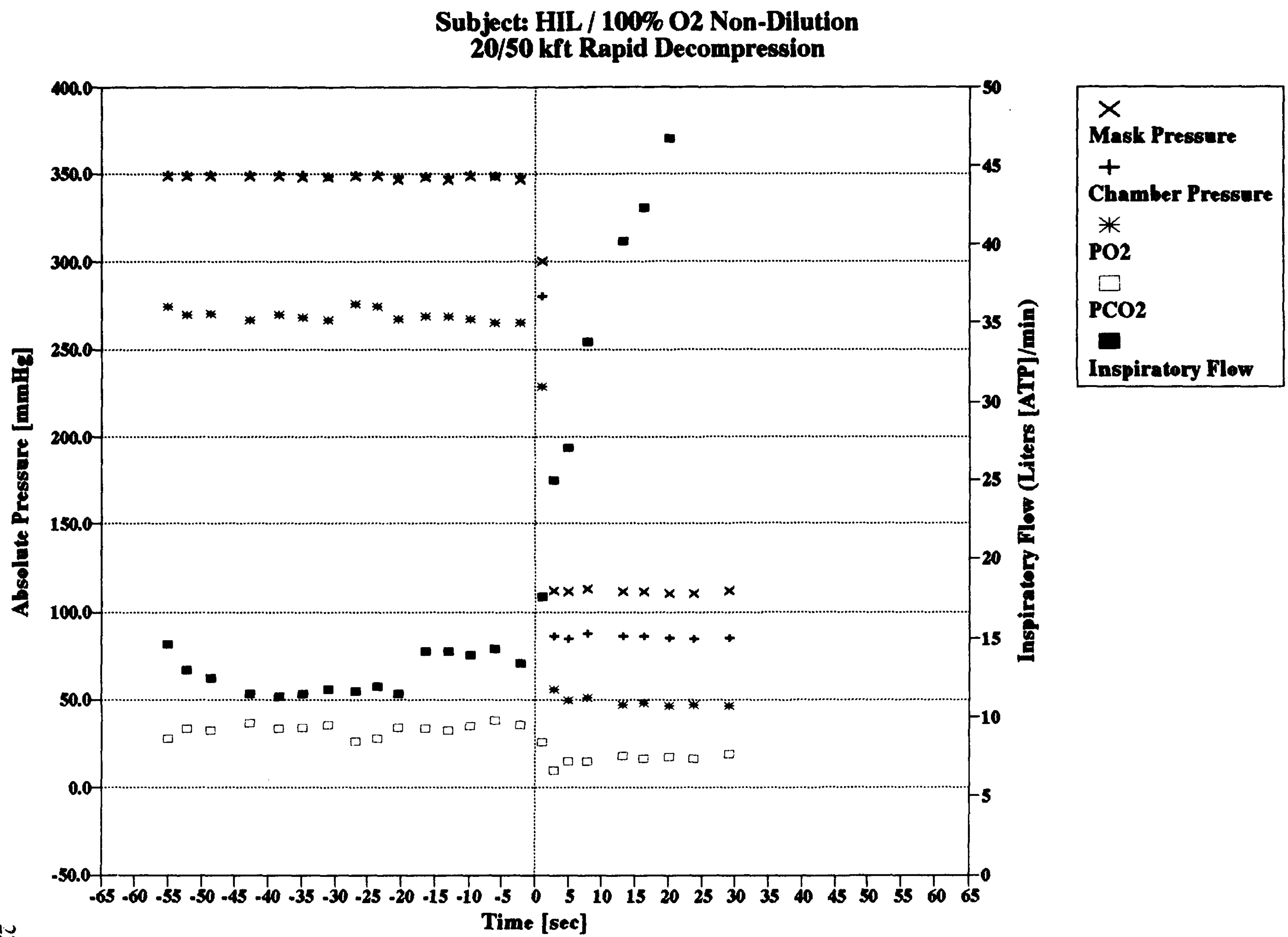


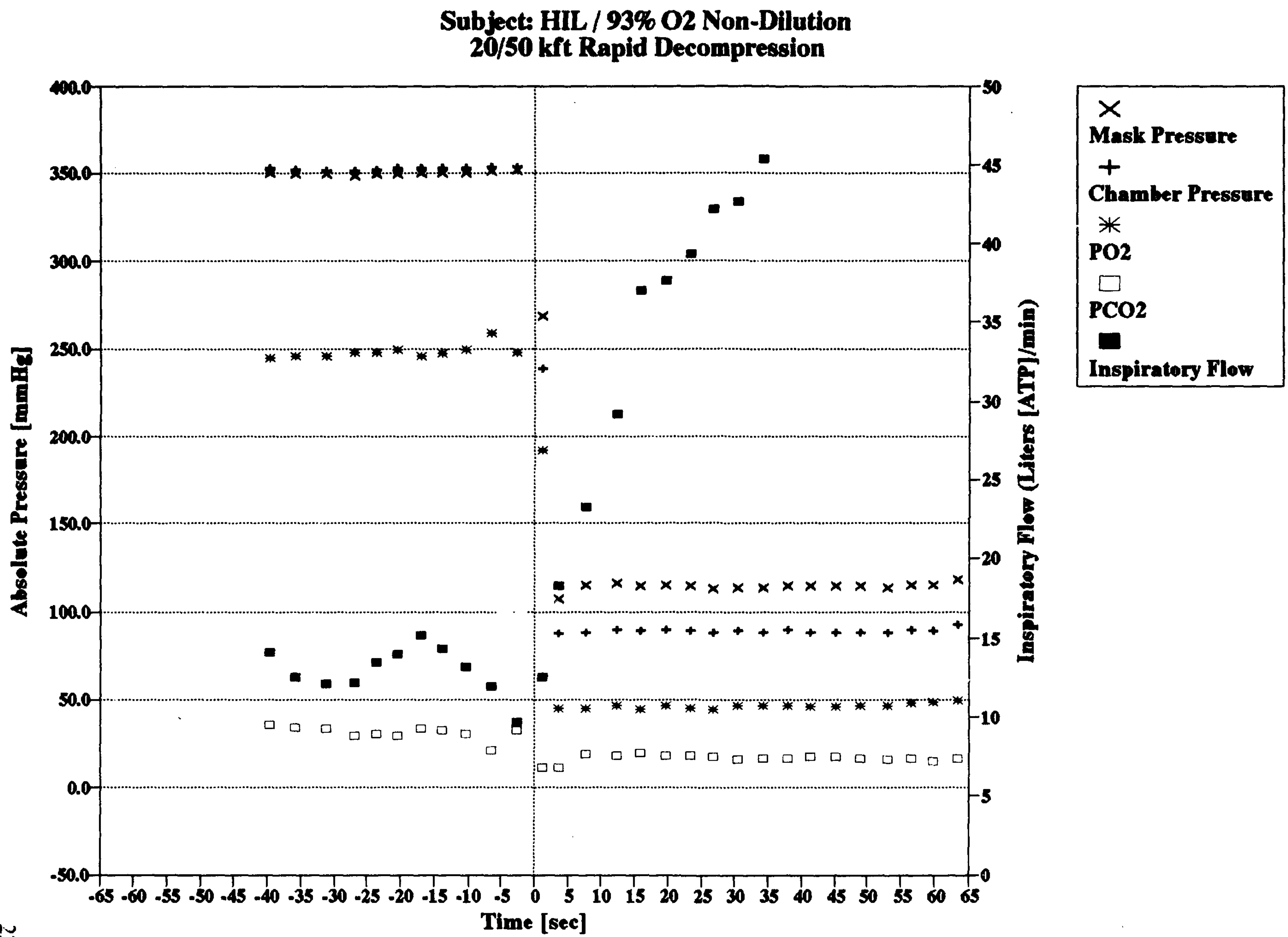
Subject: HAT / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



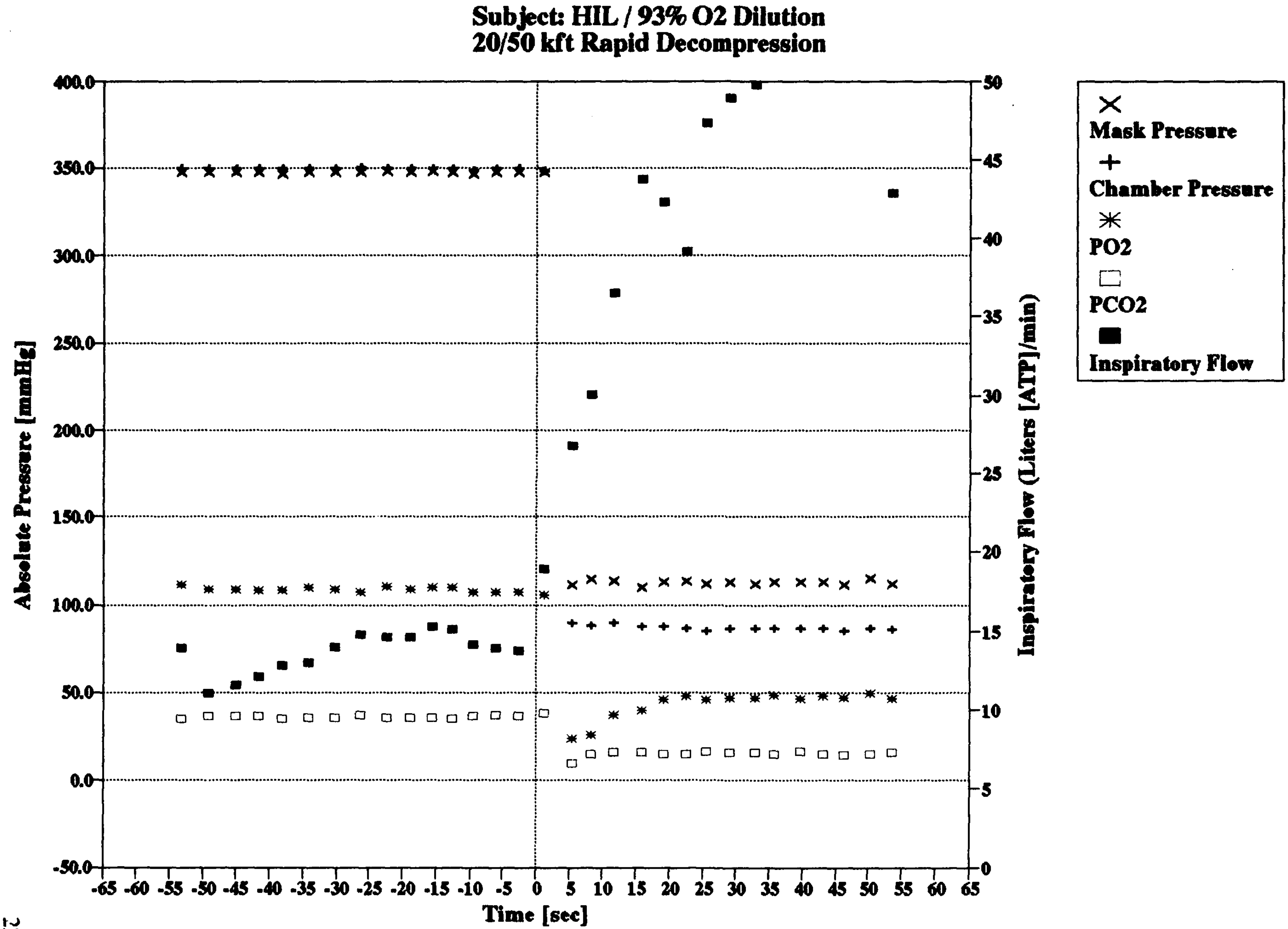


Subject: HAT / 100% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow


Subject: HIL / 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

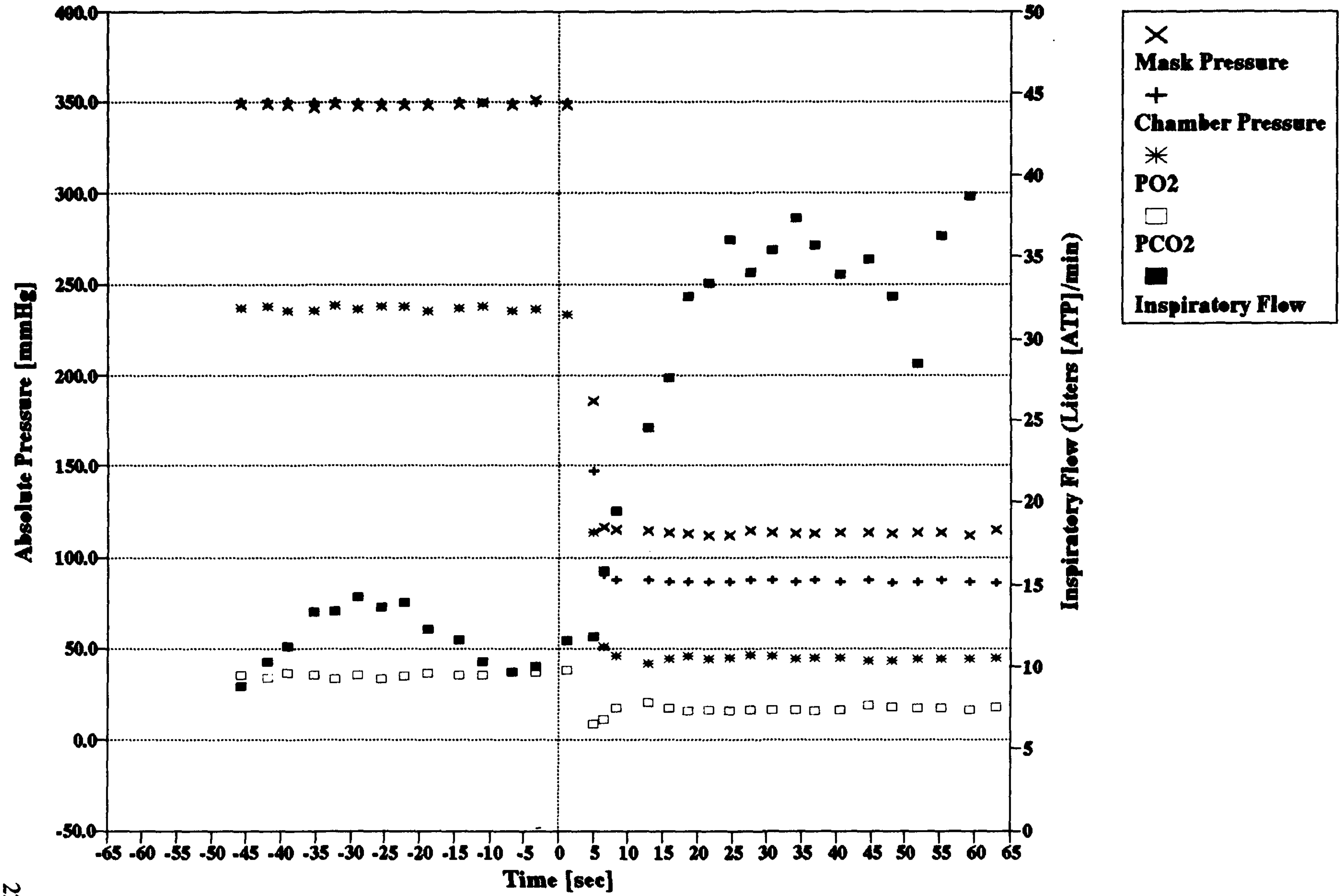


Subject: HIL / 93% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65

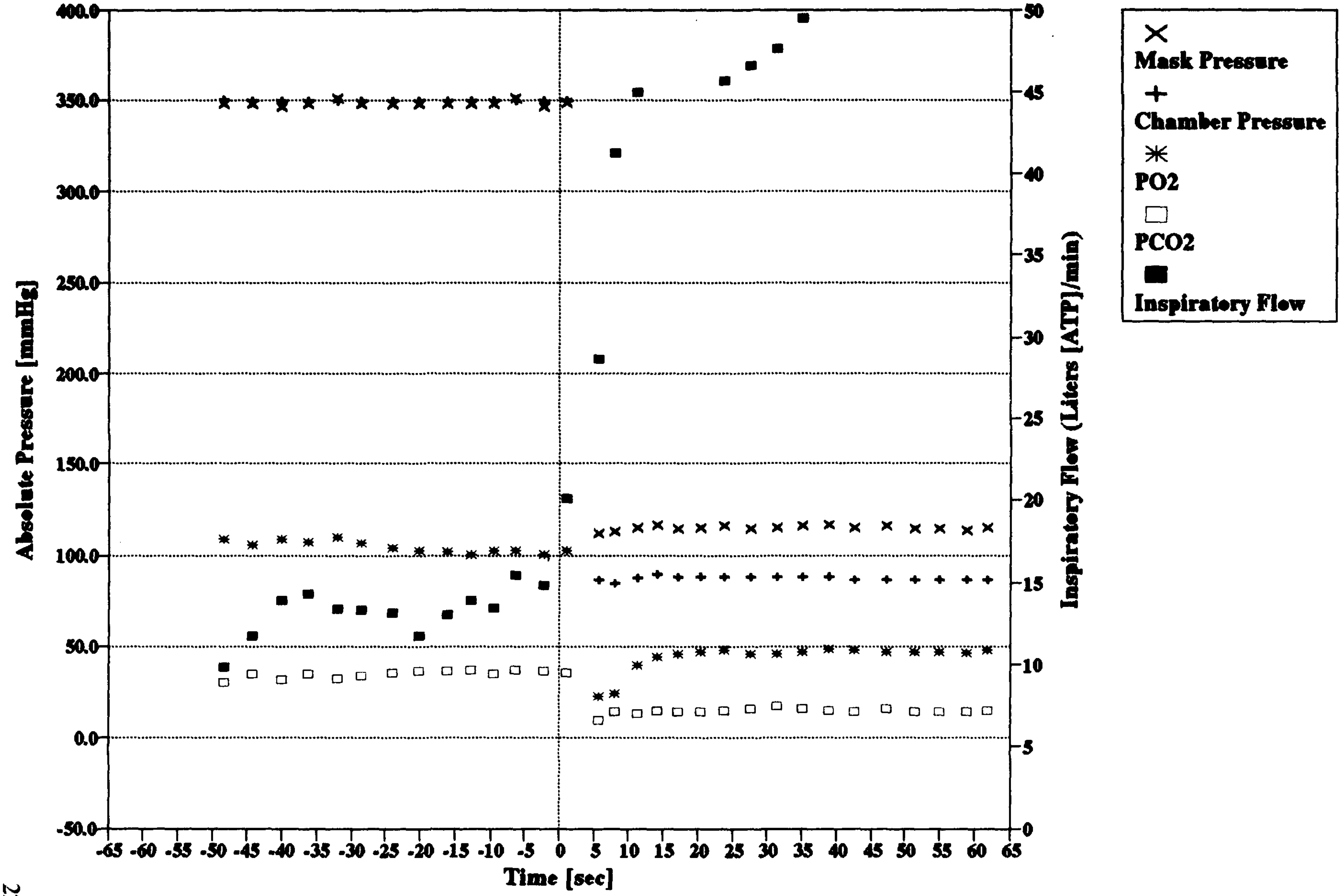


Subject: HIL / 93% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

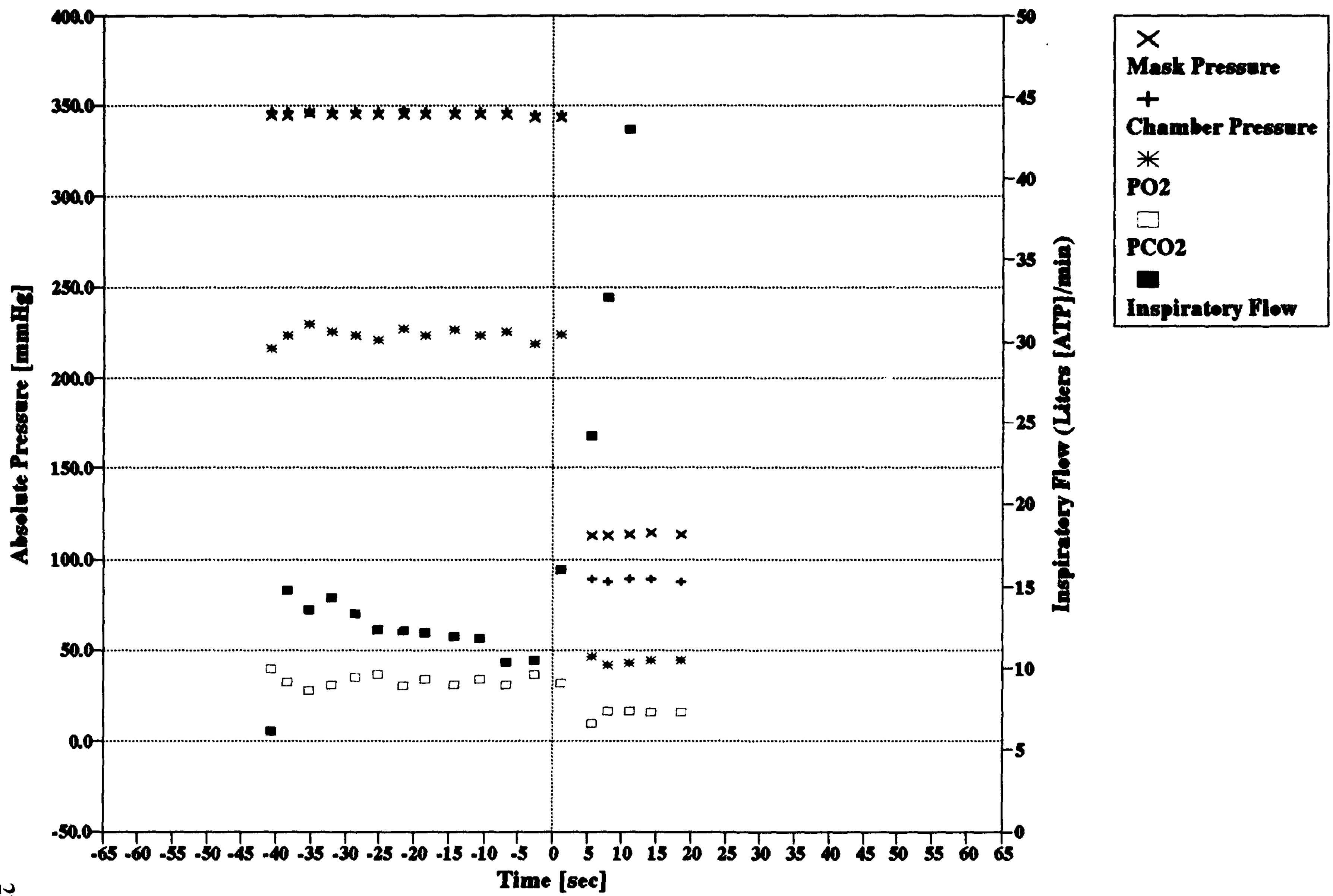
Subject: HIL / 90% O2 Non-Dilution
20/50 kft Rapid Decompression



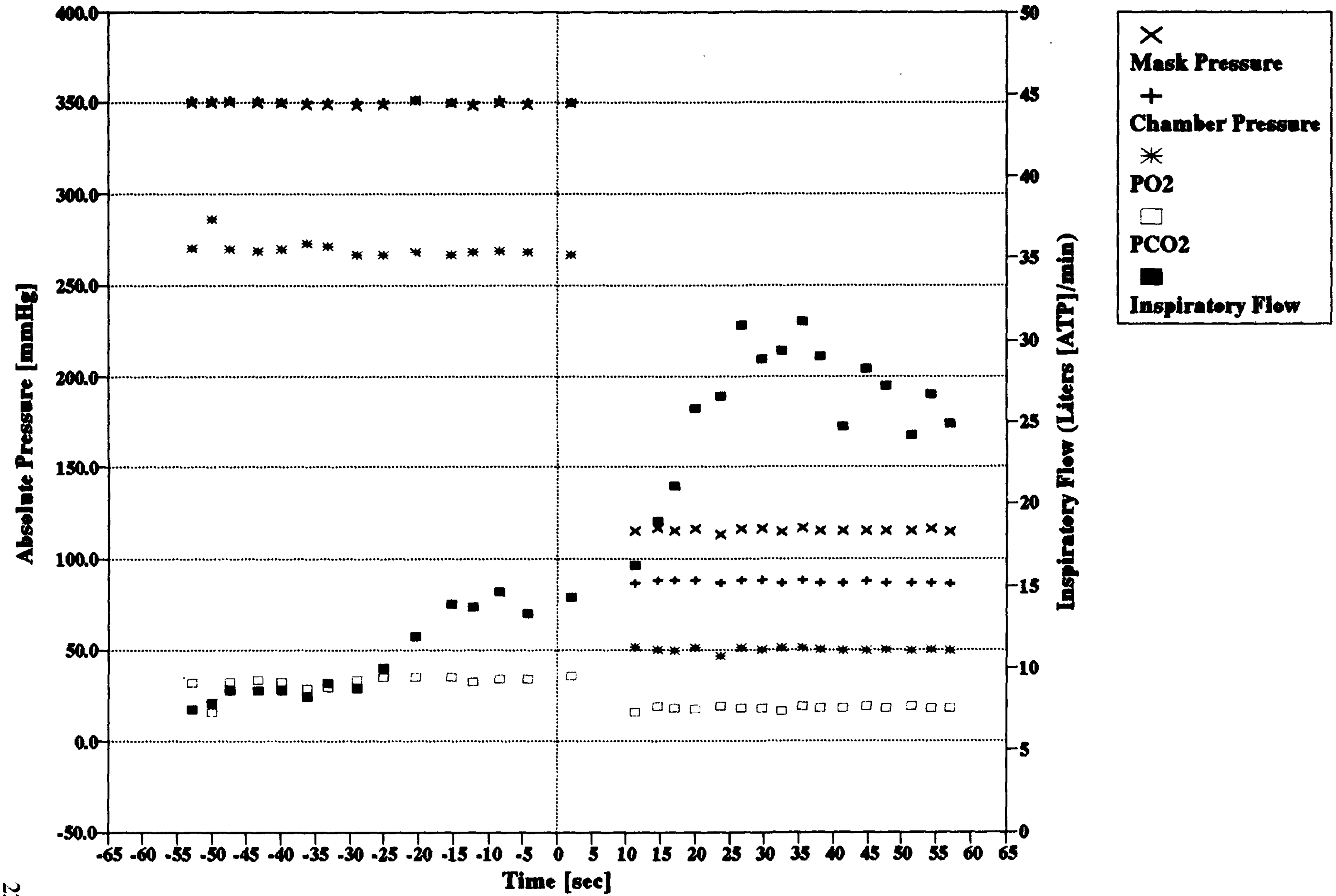
Subject: HIL / 90% O2 Dilution
20/50 kft Rapid Decompression



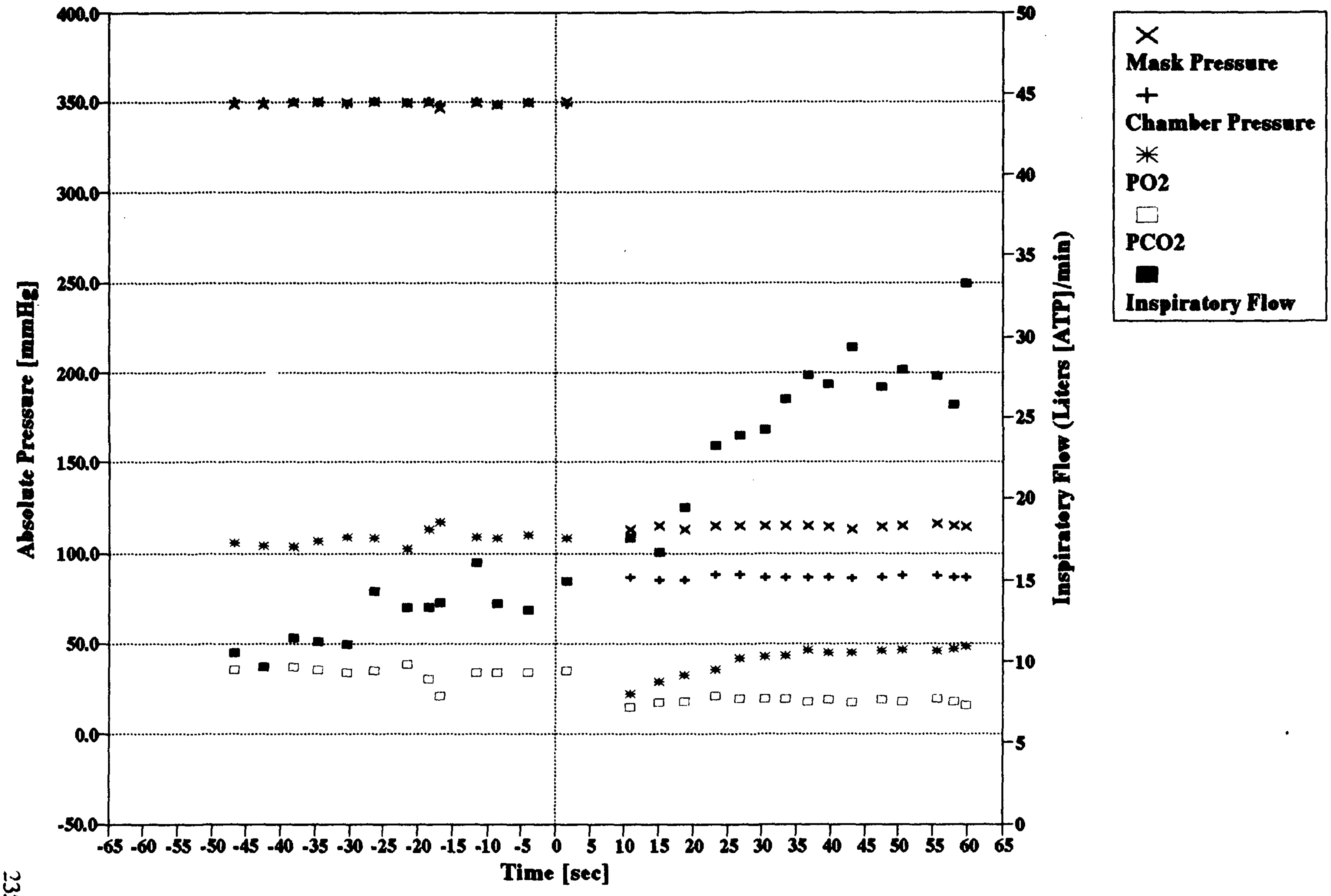
Subject: HIL / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



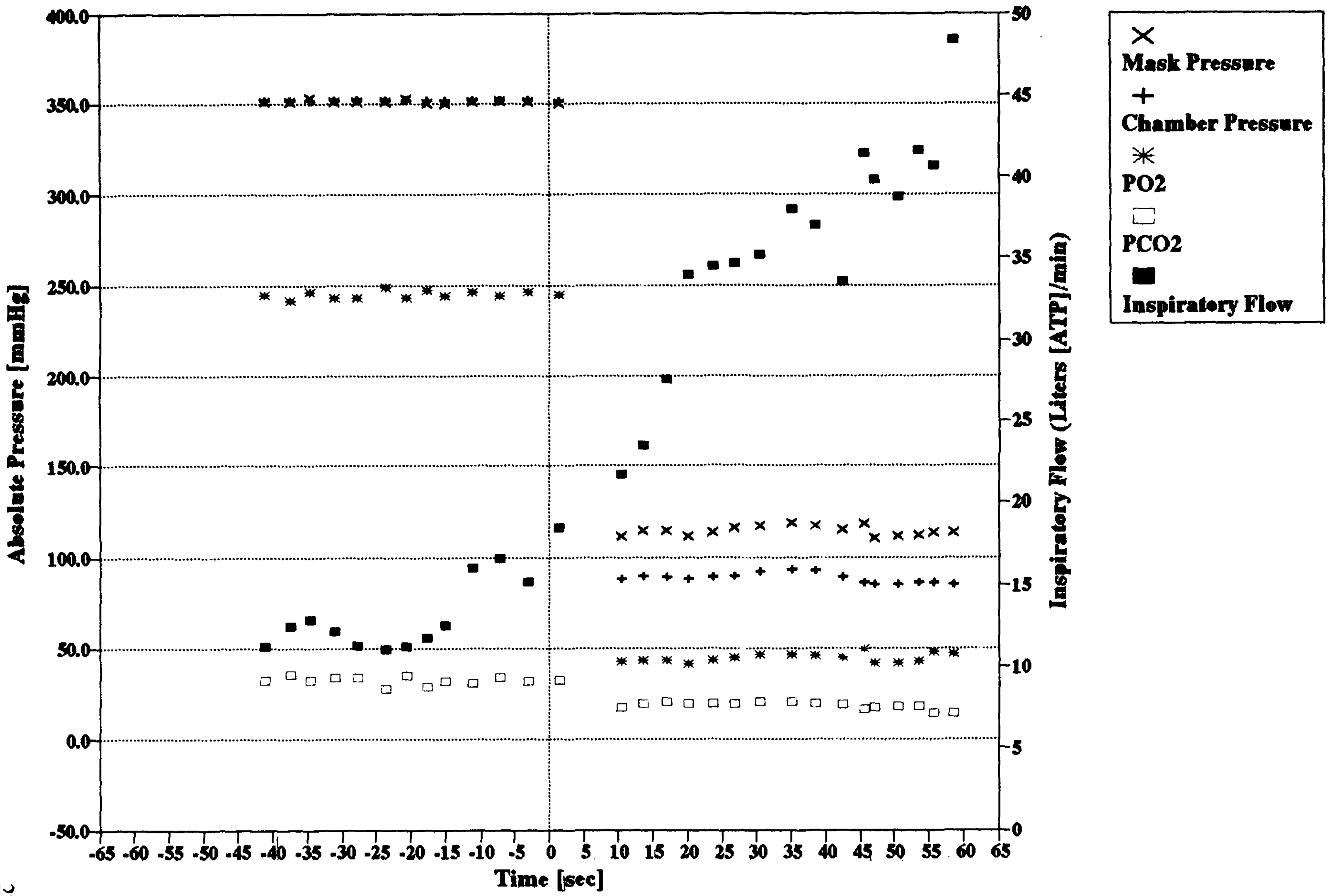
Subject: HOS / 100% O2 Non-Dilution
20/50 kft Rapid Decompression

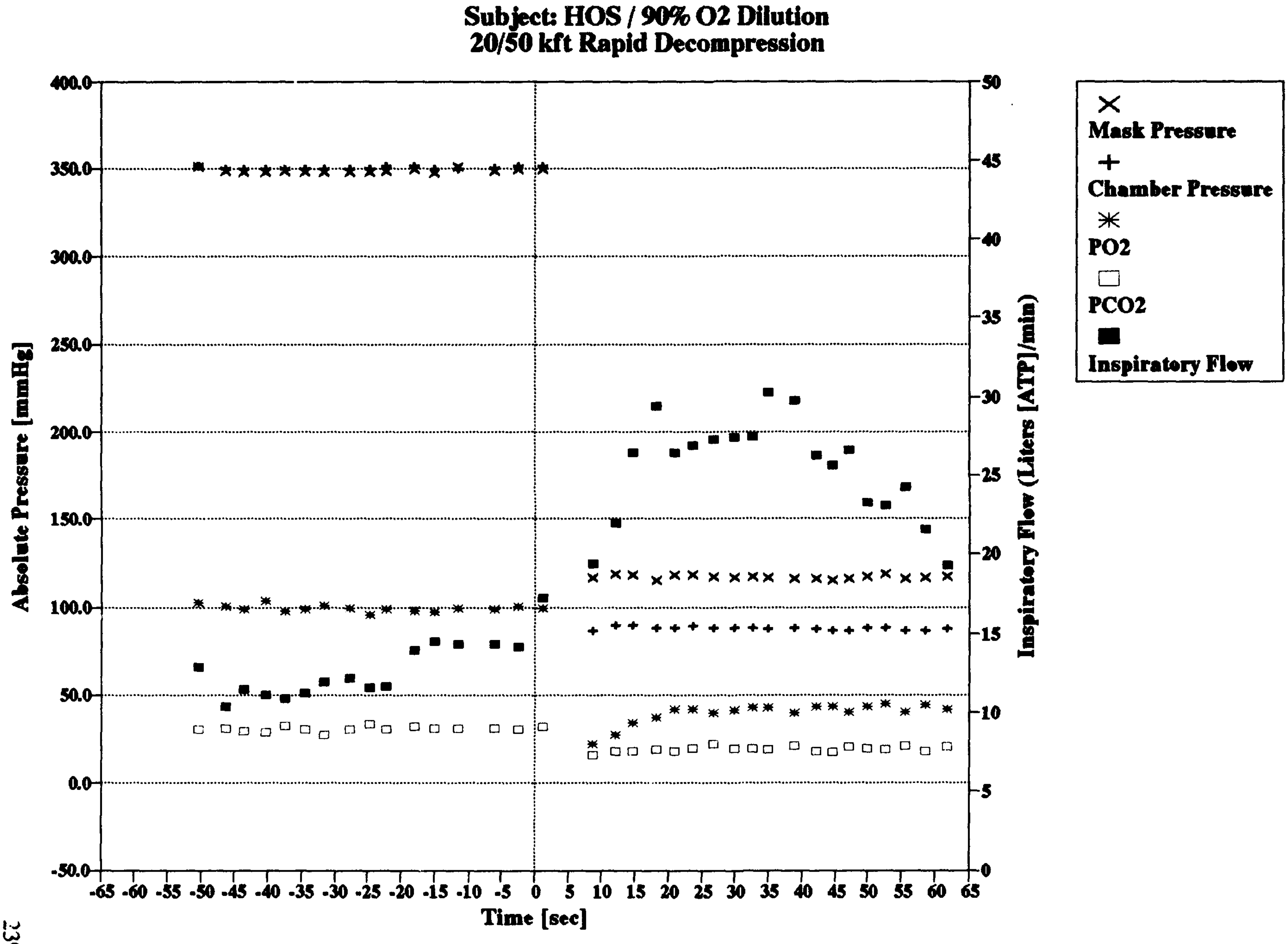


Subject: HOS / 93% O2 Dilution
20/50 kft Rapid Decompression

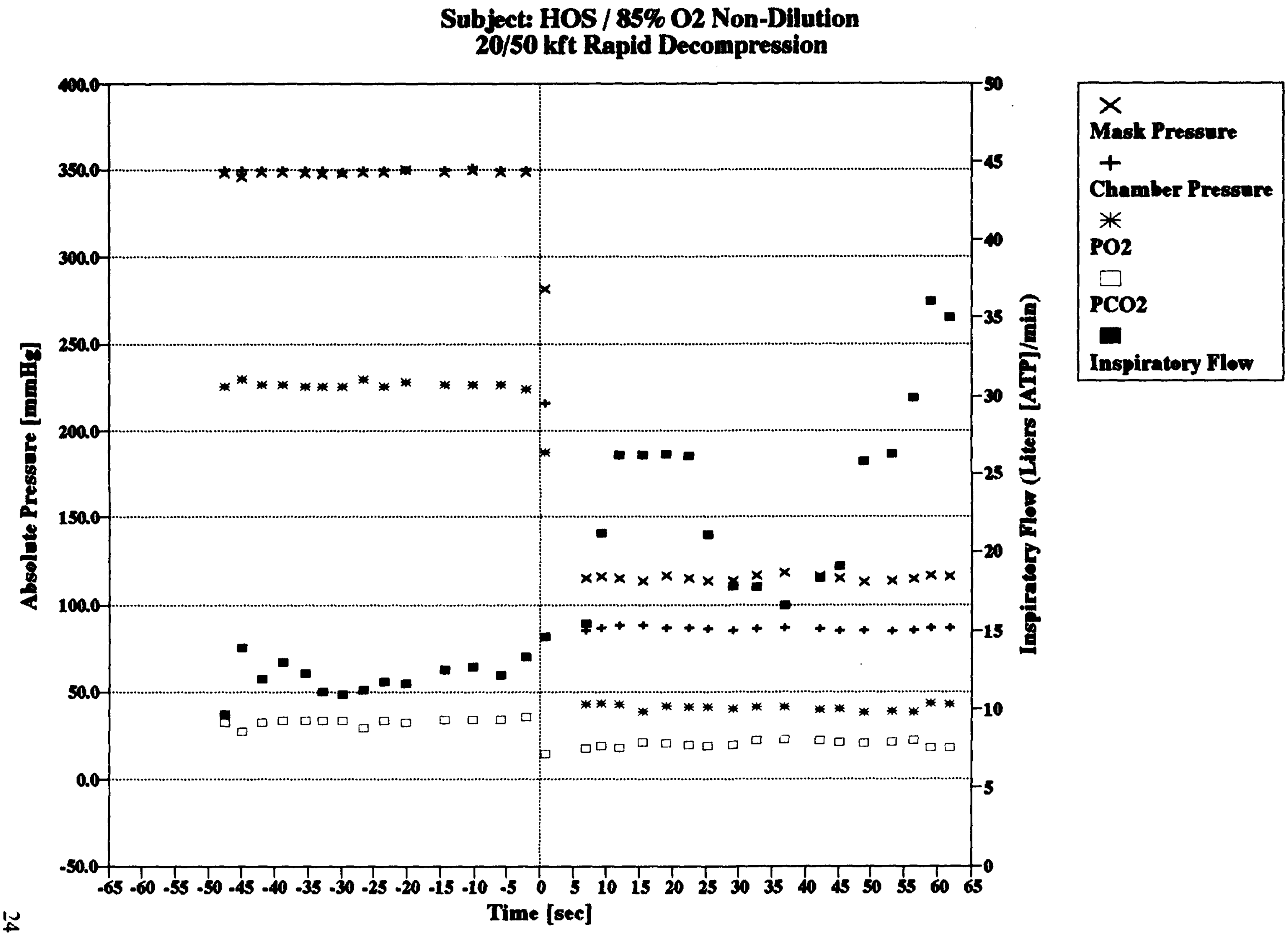


Subject: HOS / 90% O2 Non-Dilution
20/50 kft Rapid Decompression



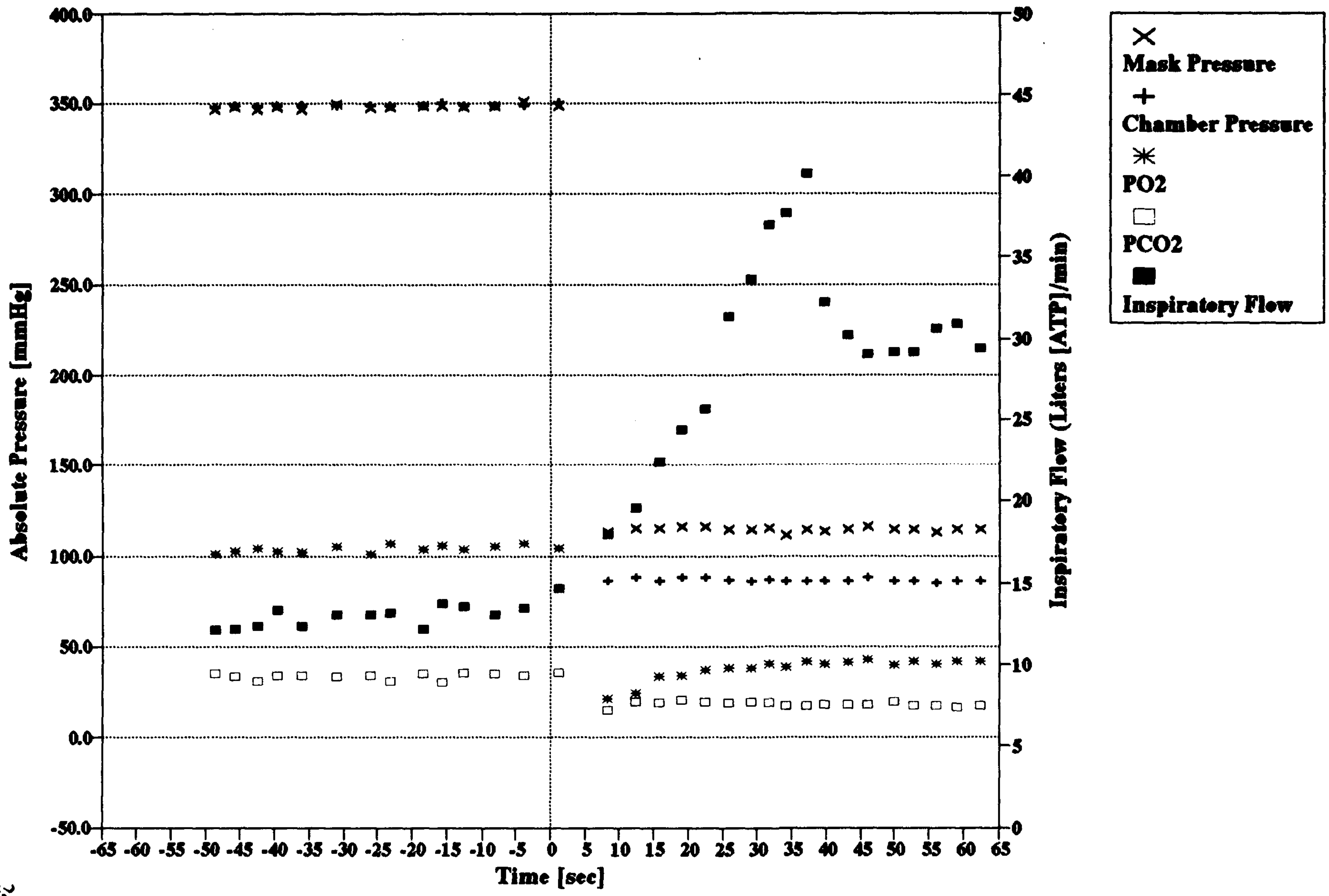


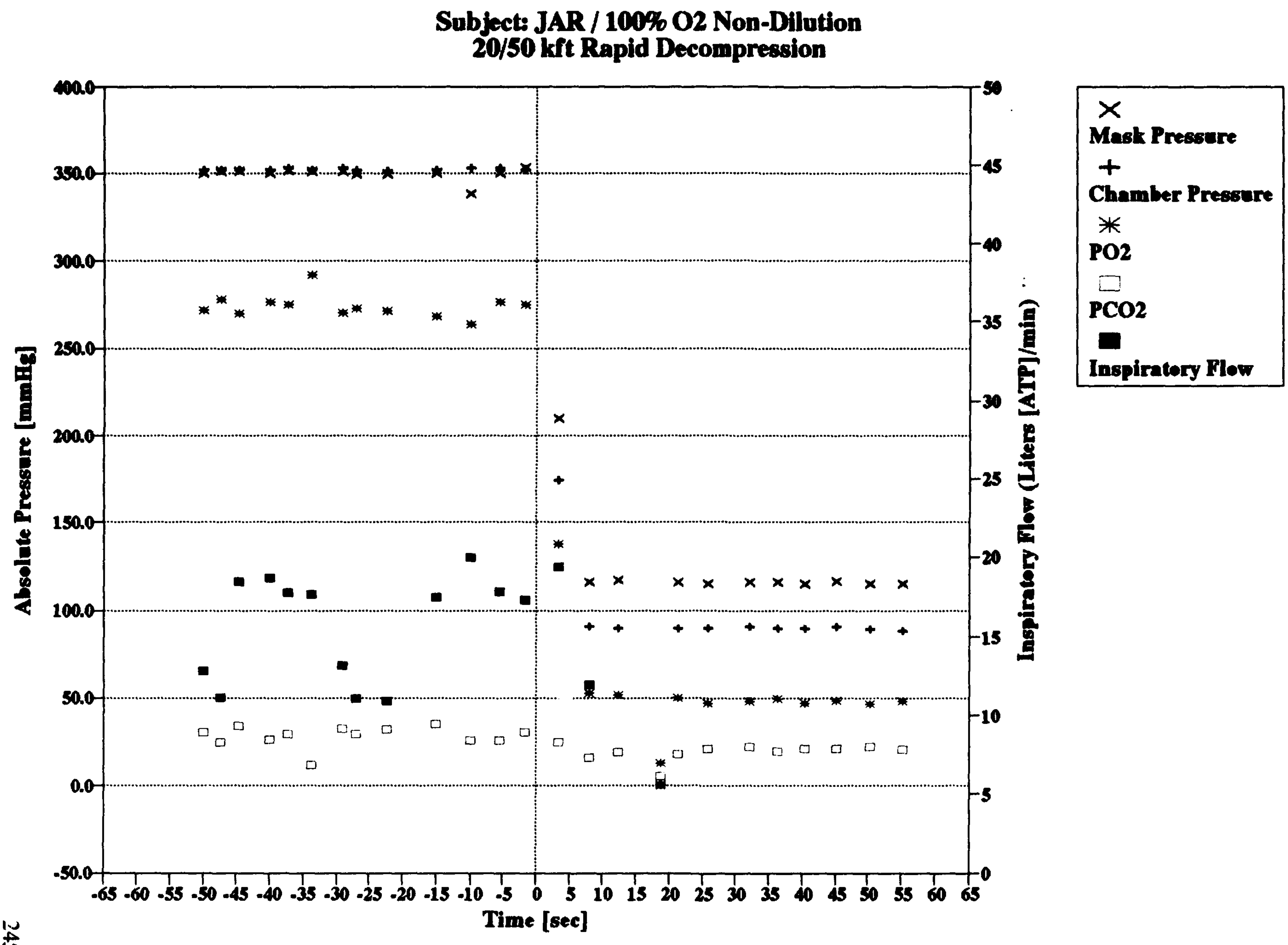
Subject: HOS / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow



Subject: HOS / 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

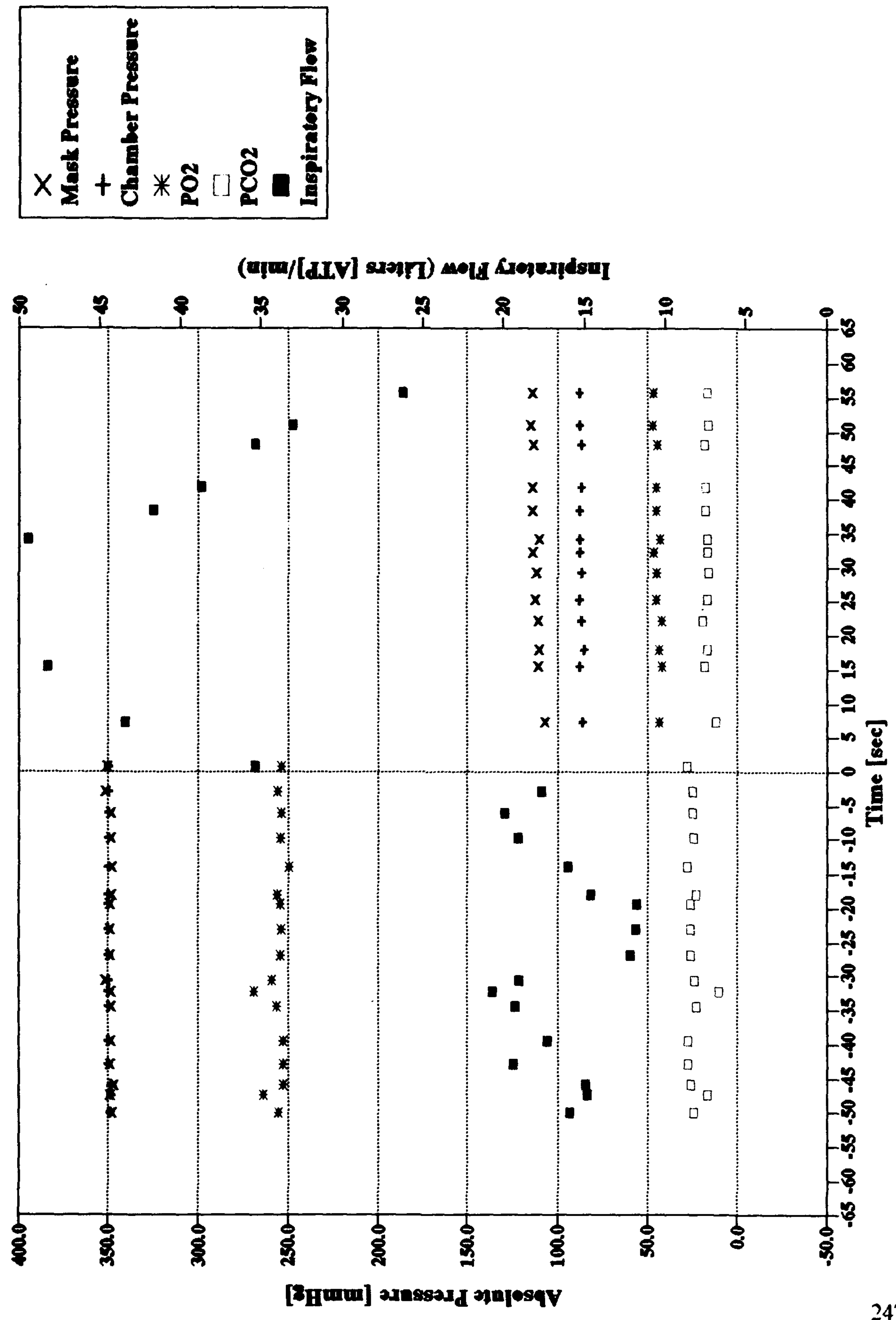
Subject: HOS / 85% O2 Dilution
20/50 kft Rapid Decompression



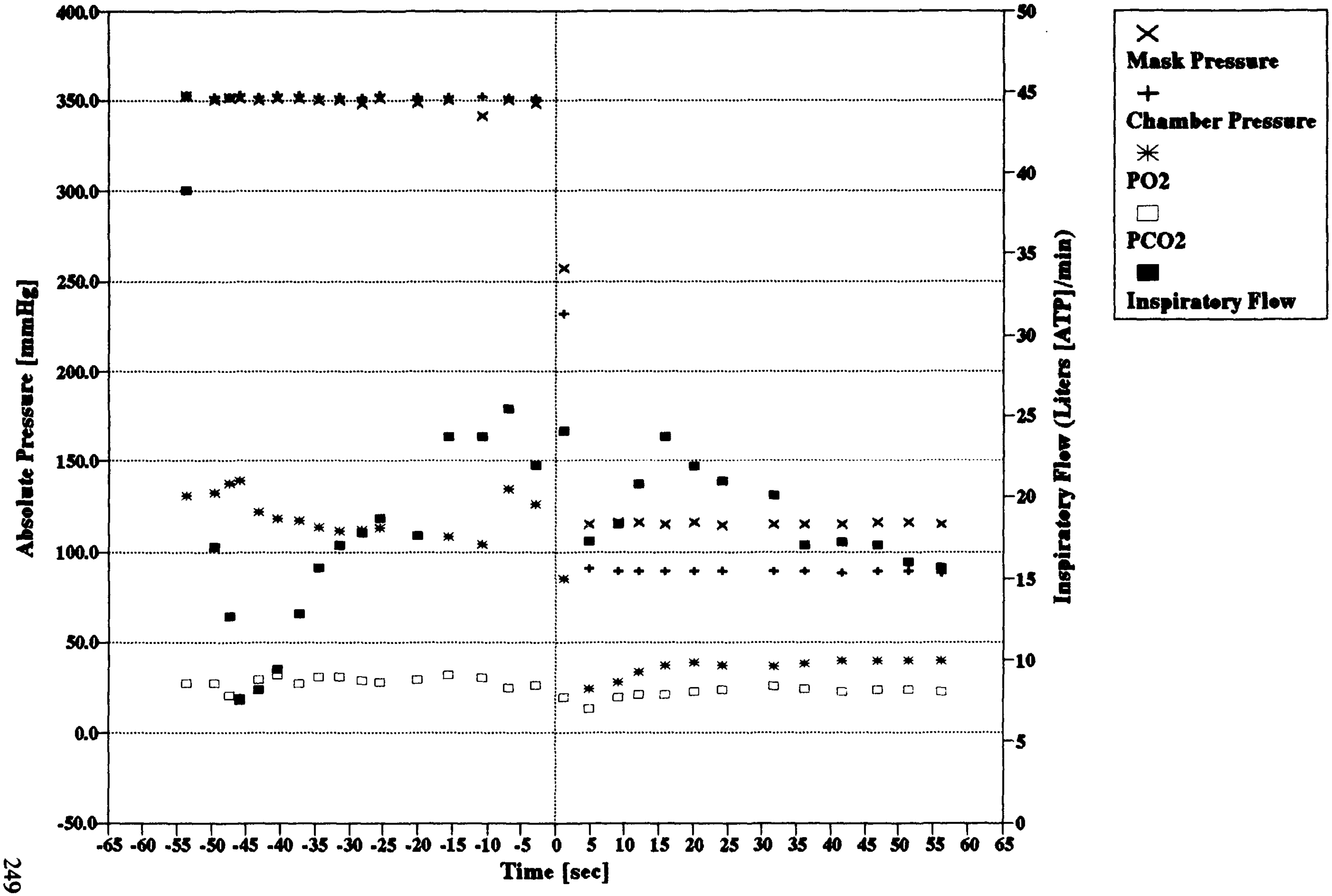


Subject: JAR / 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

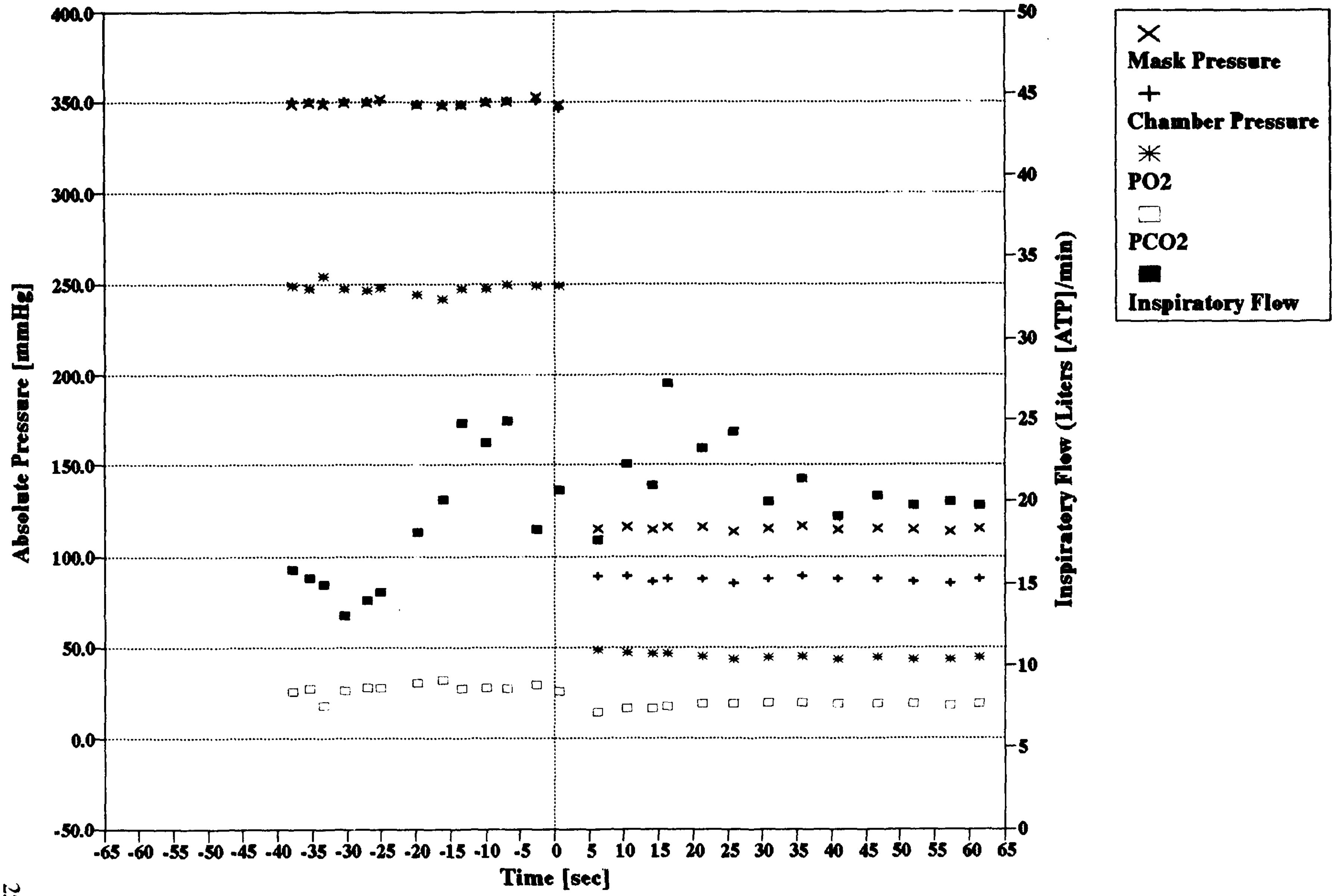
Subject: JAR / 93% O2 Non-Dilution
20/50 kft Rapid Decompression

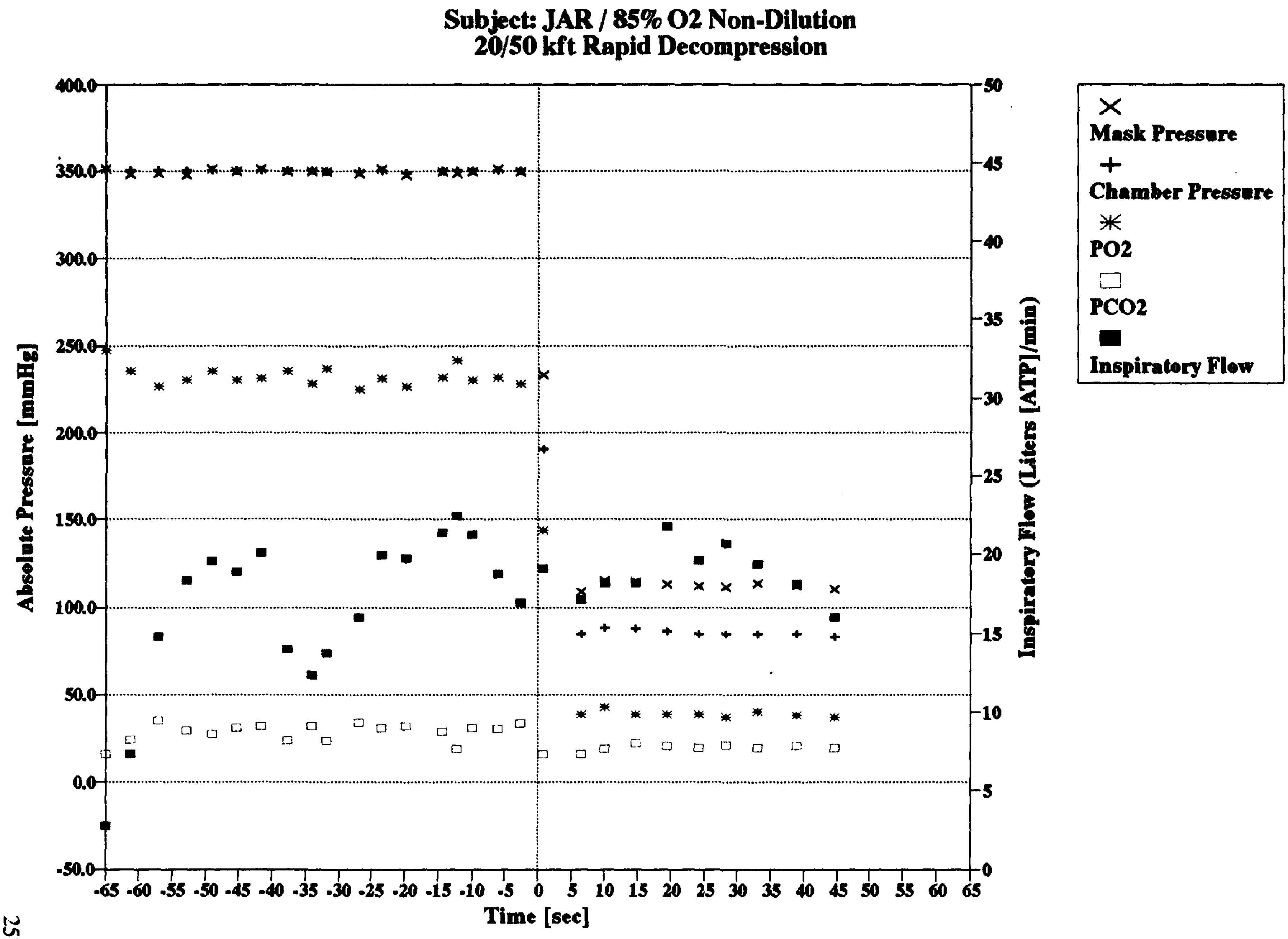


Subject: JAR / 93% O2 Dilution
20/50 kft Rapid Decompression



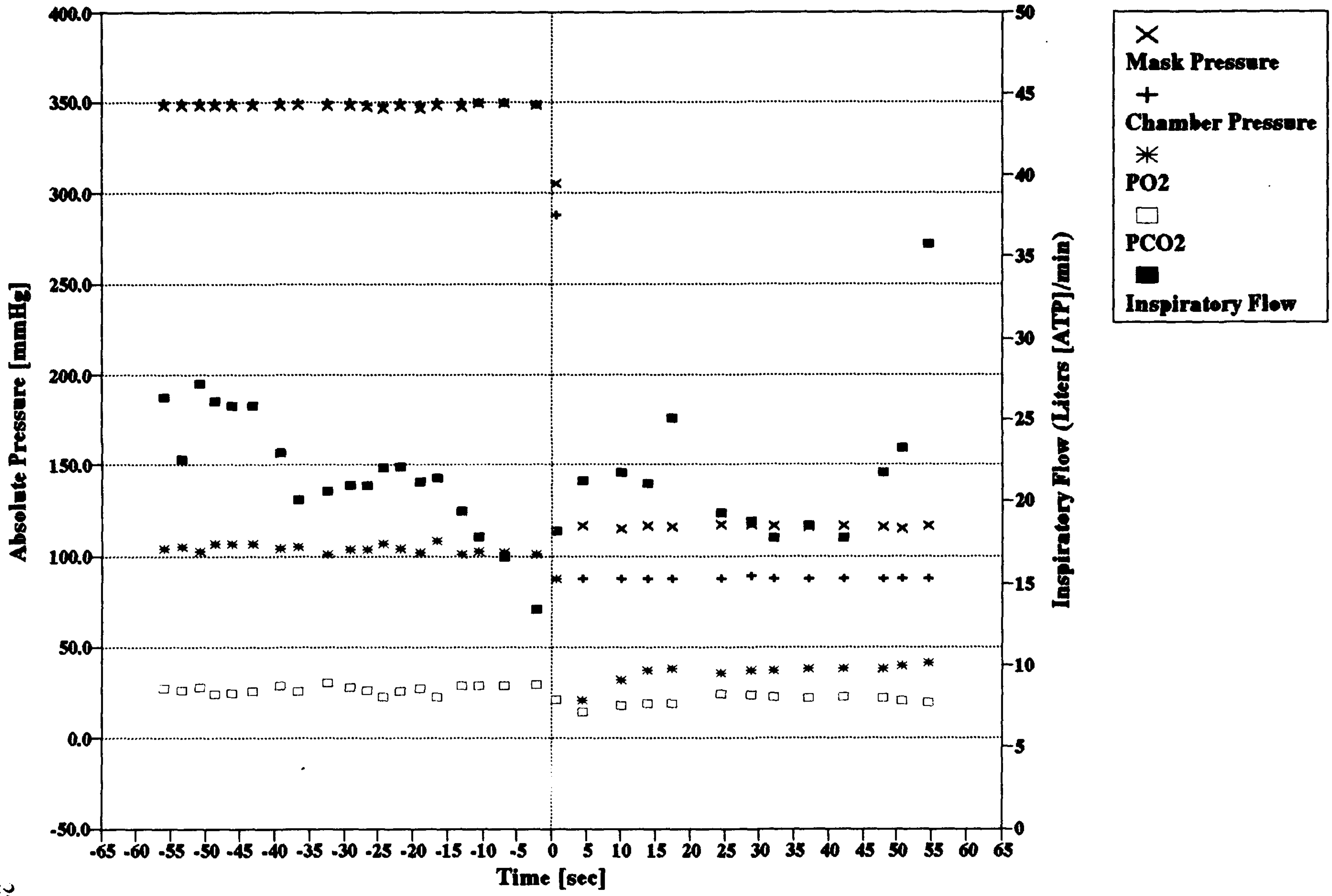
Subject: JAR / 90% O2 Non-Dilution
20/50 kft Rapid Decompression

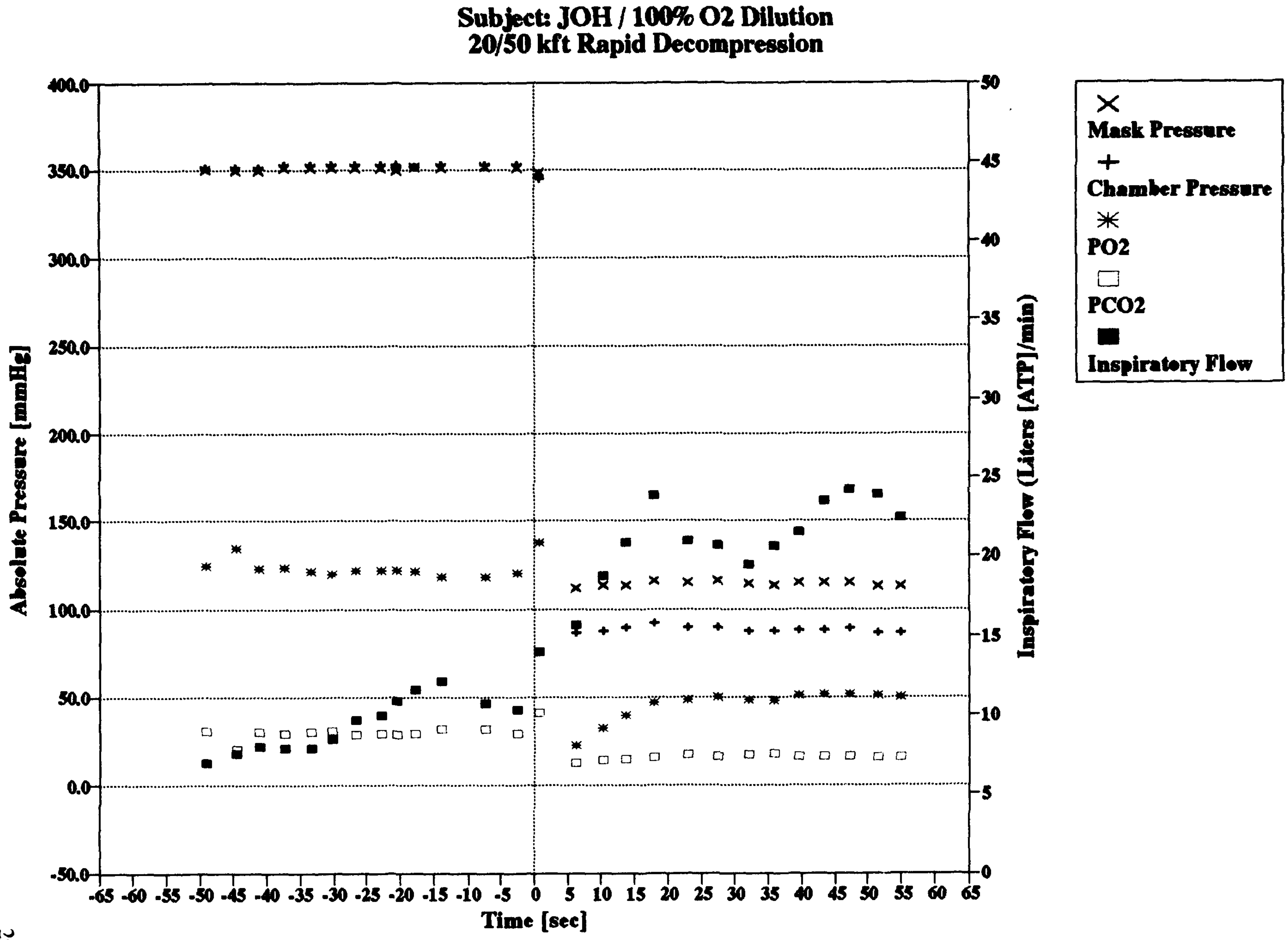




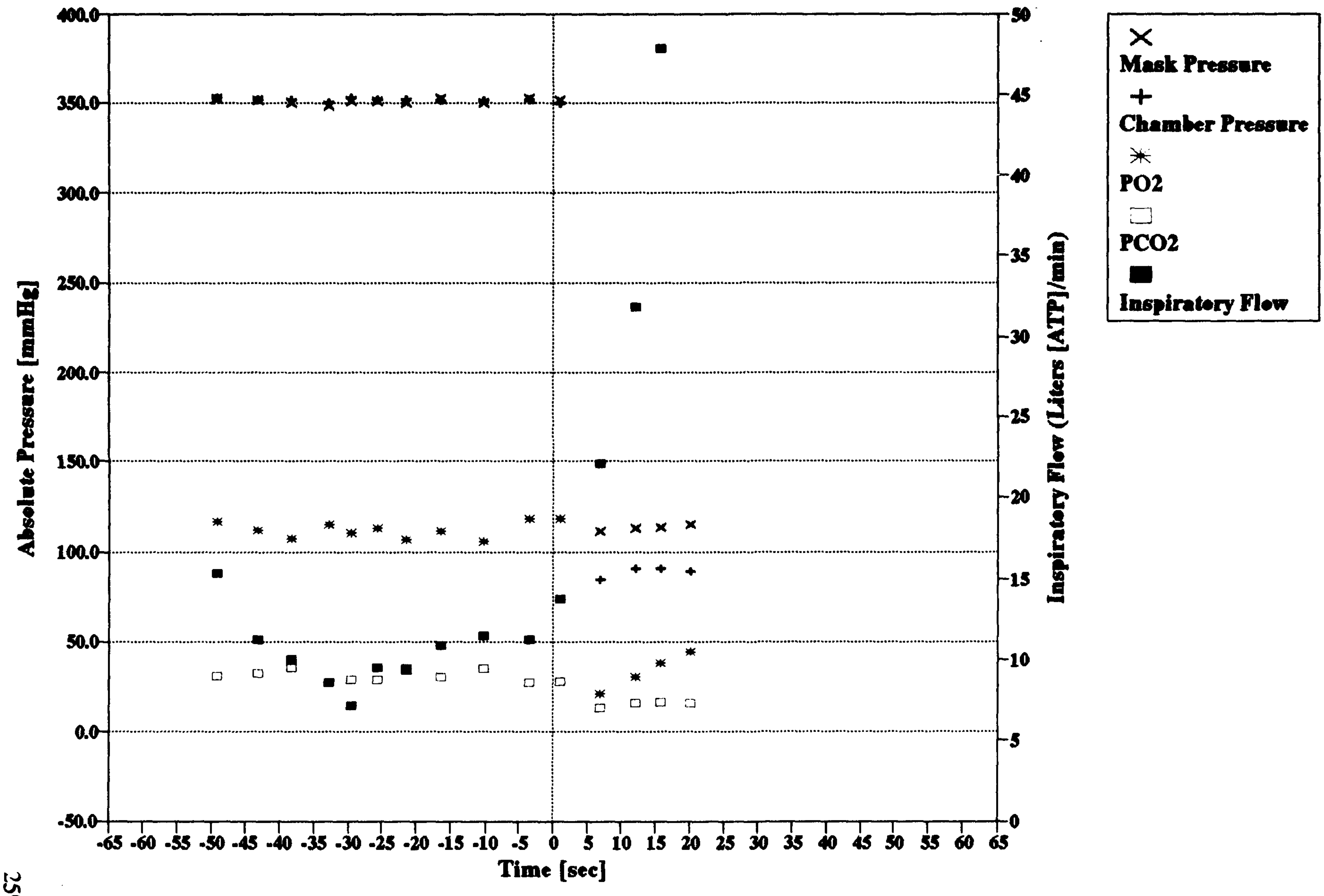
Subject: JAR / 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

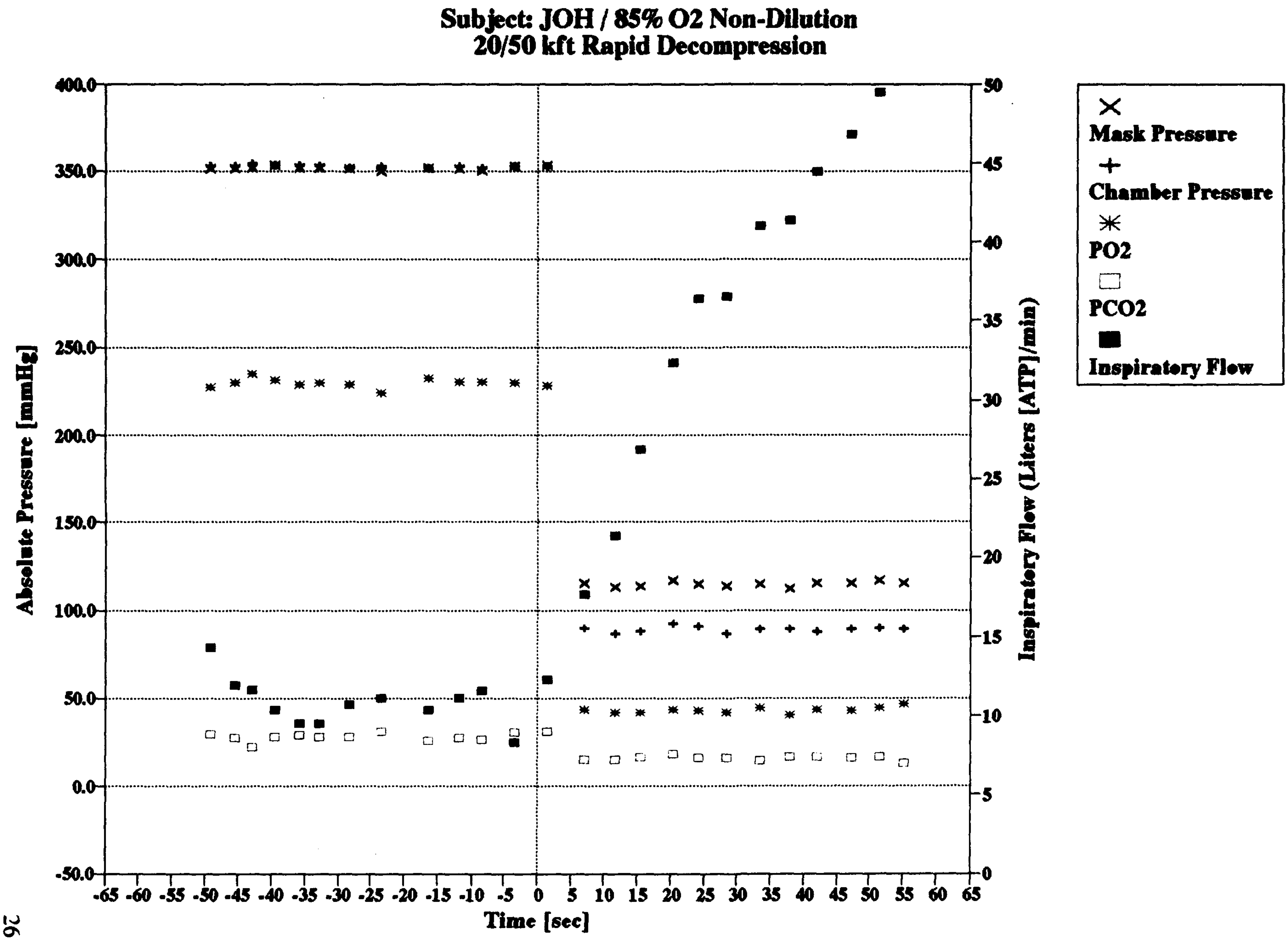
Subject: JAR / 85% O2 Dilution
20/50 kft Rapid Decompression



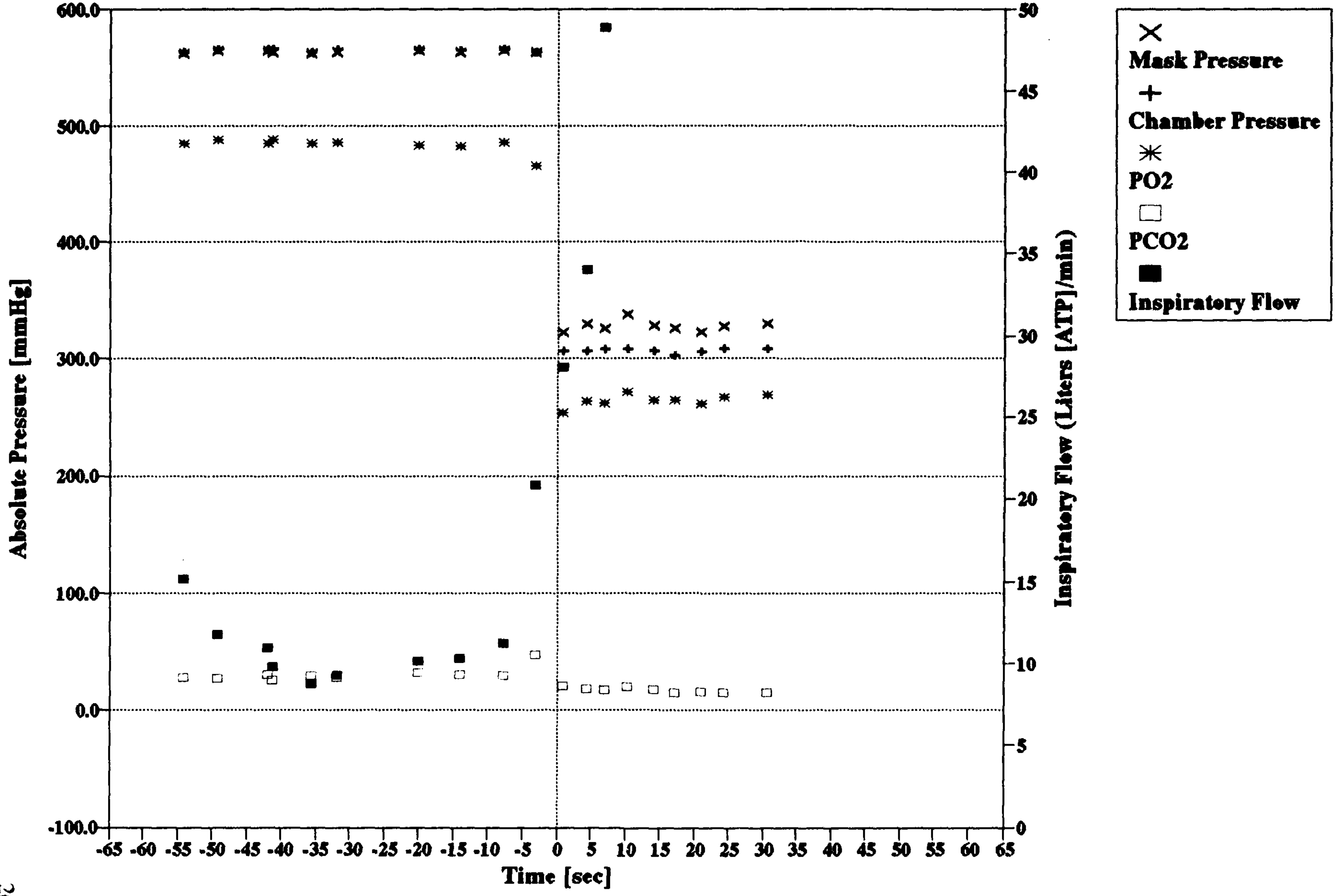

Subject: JOH / 100% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: JOH / 93% O2 Dilution
20/50 kft Rapid Decompression

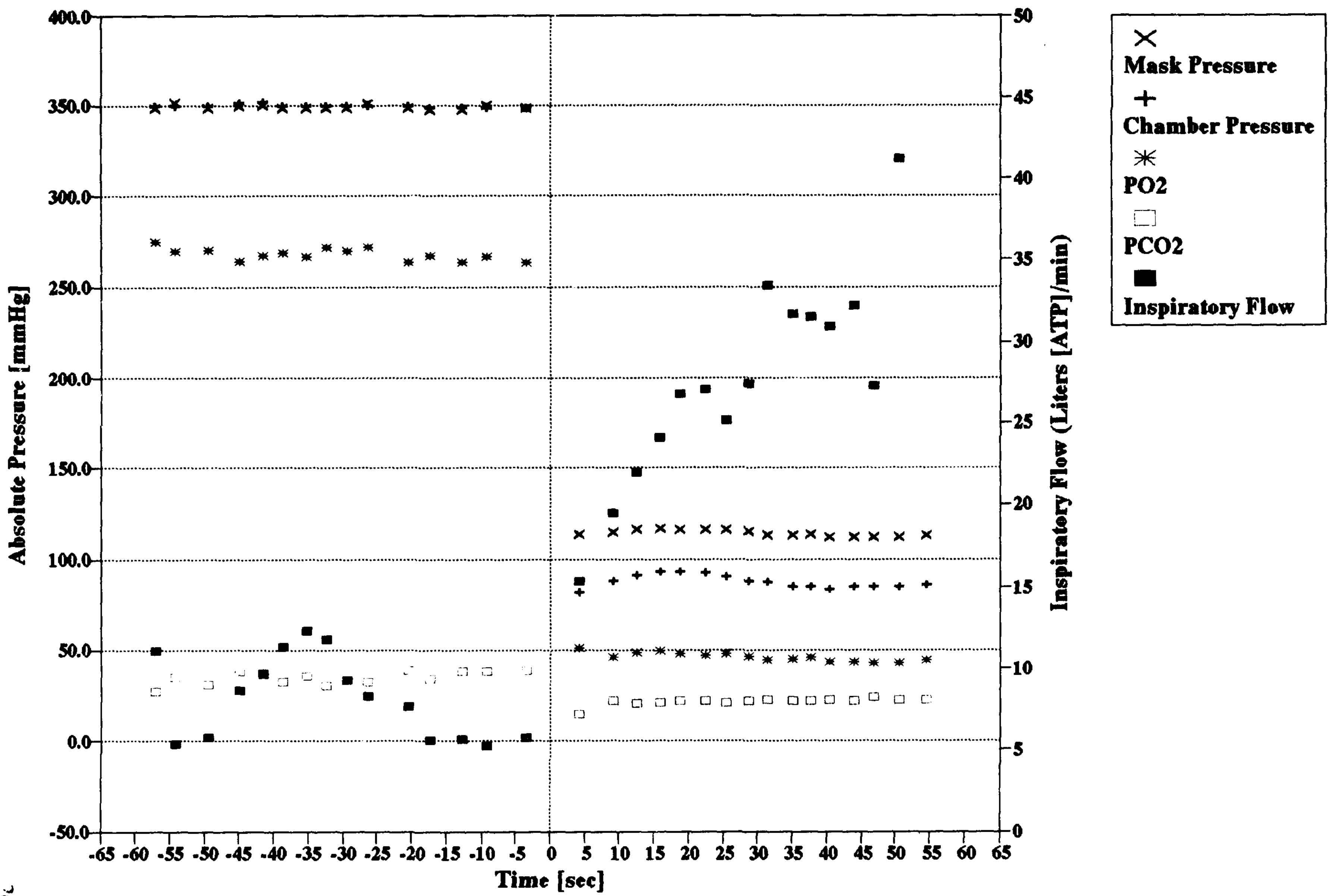


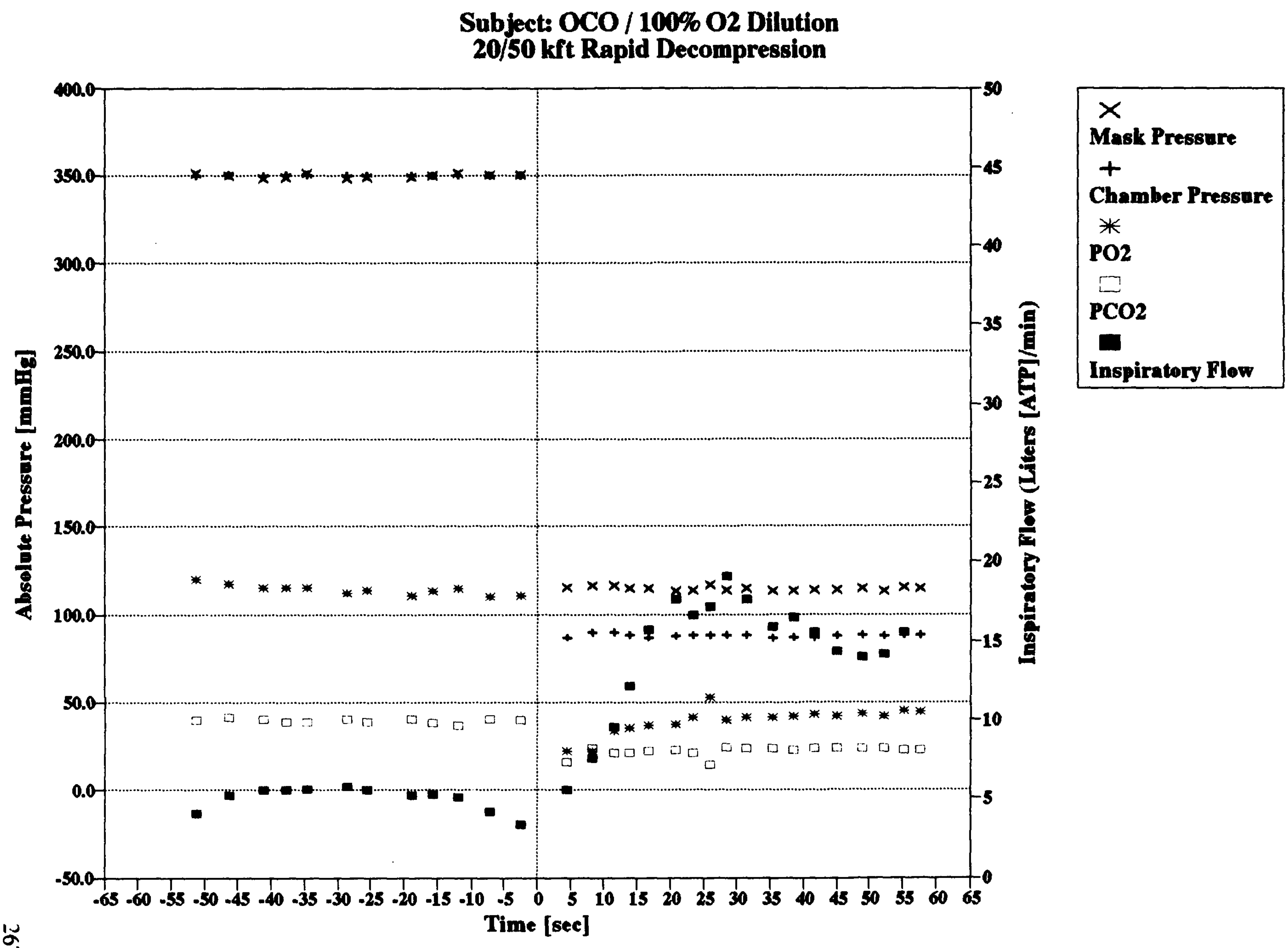

Subject: JOH / 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: JOH / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



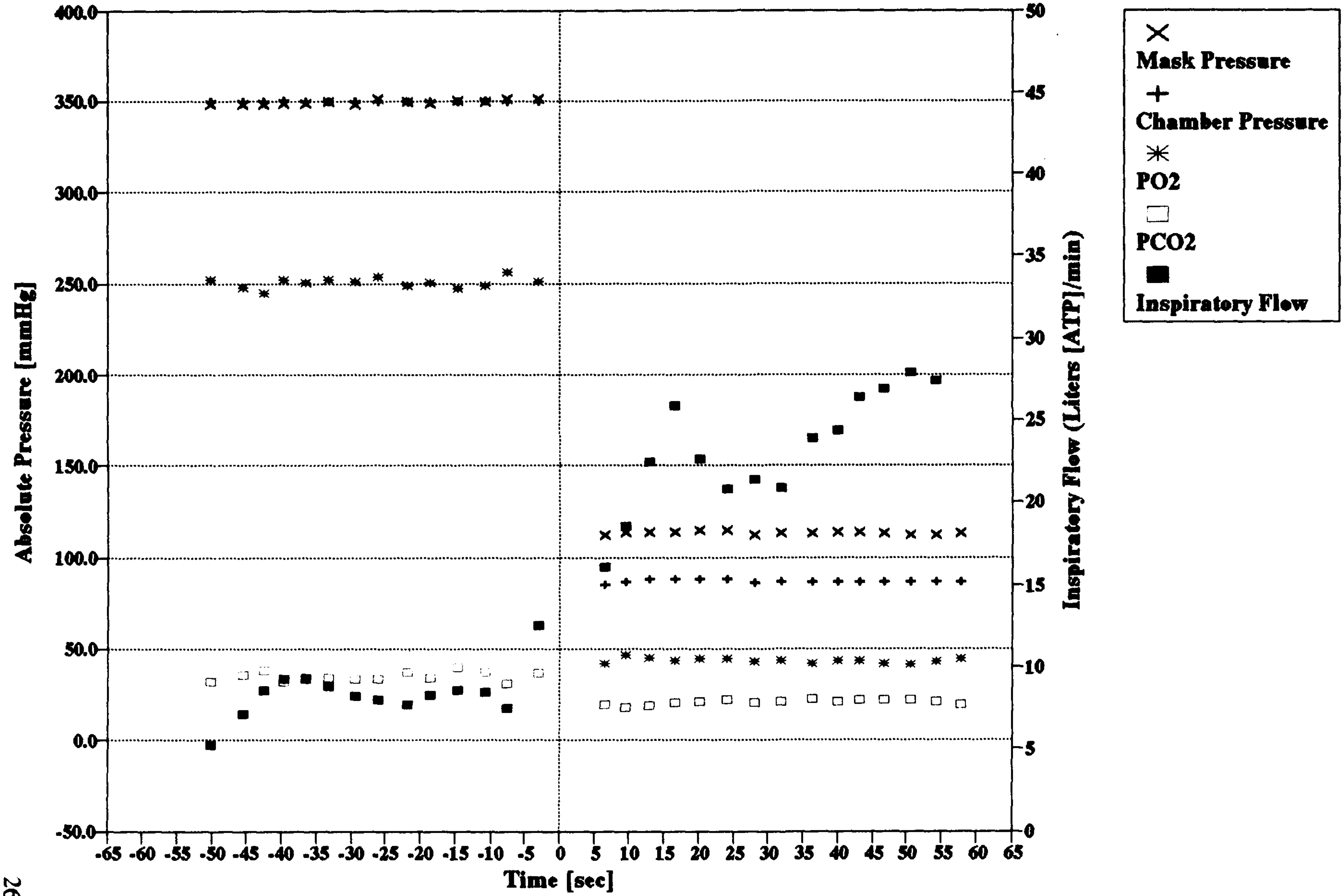
Subject: OCO / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



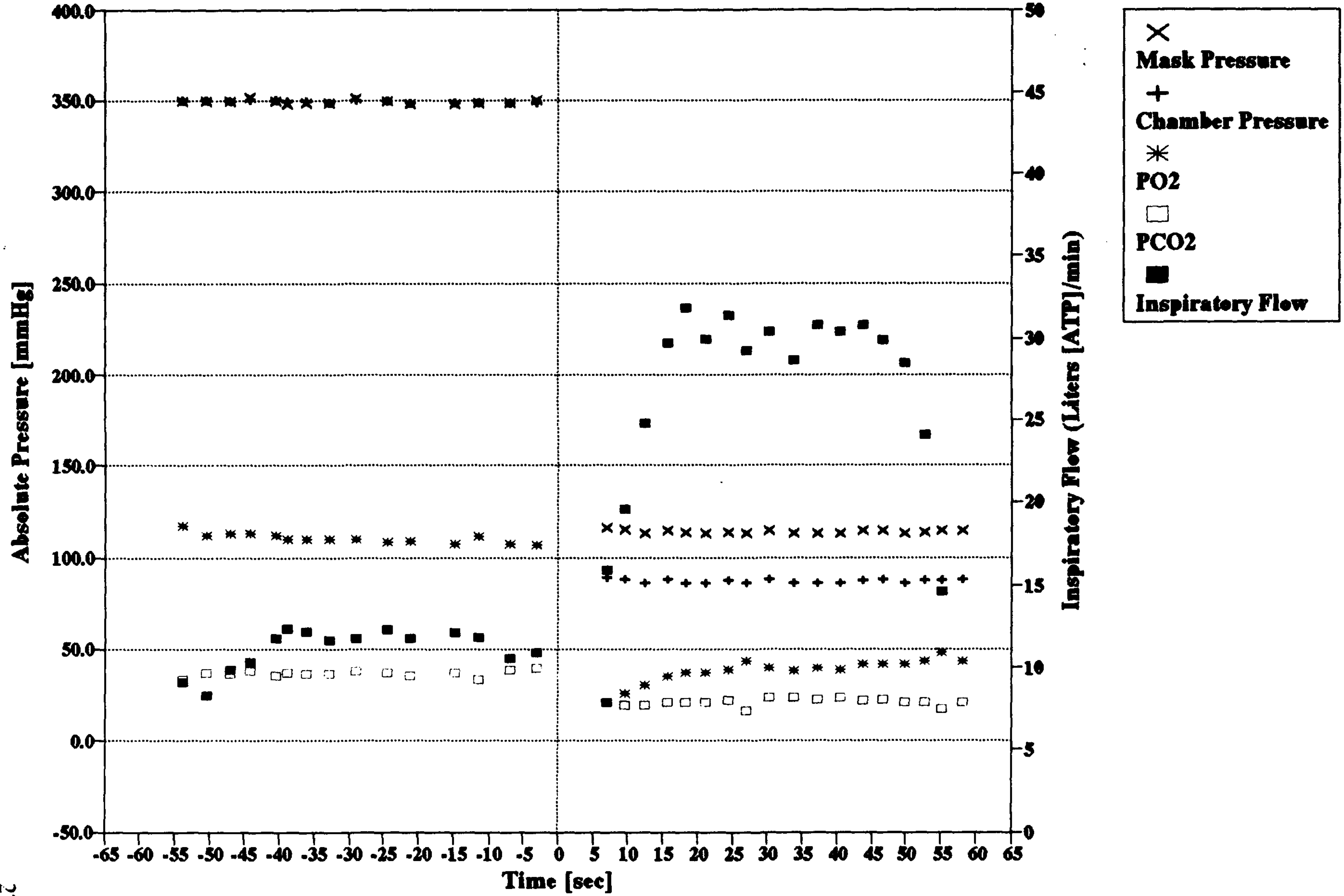


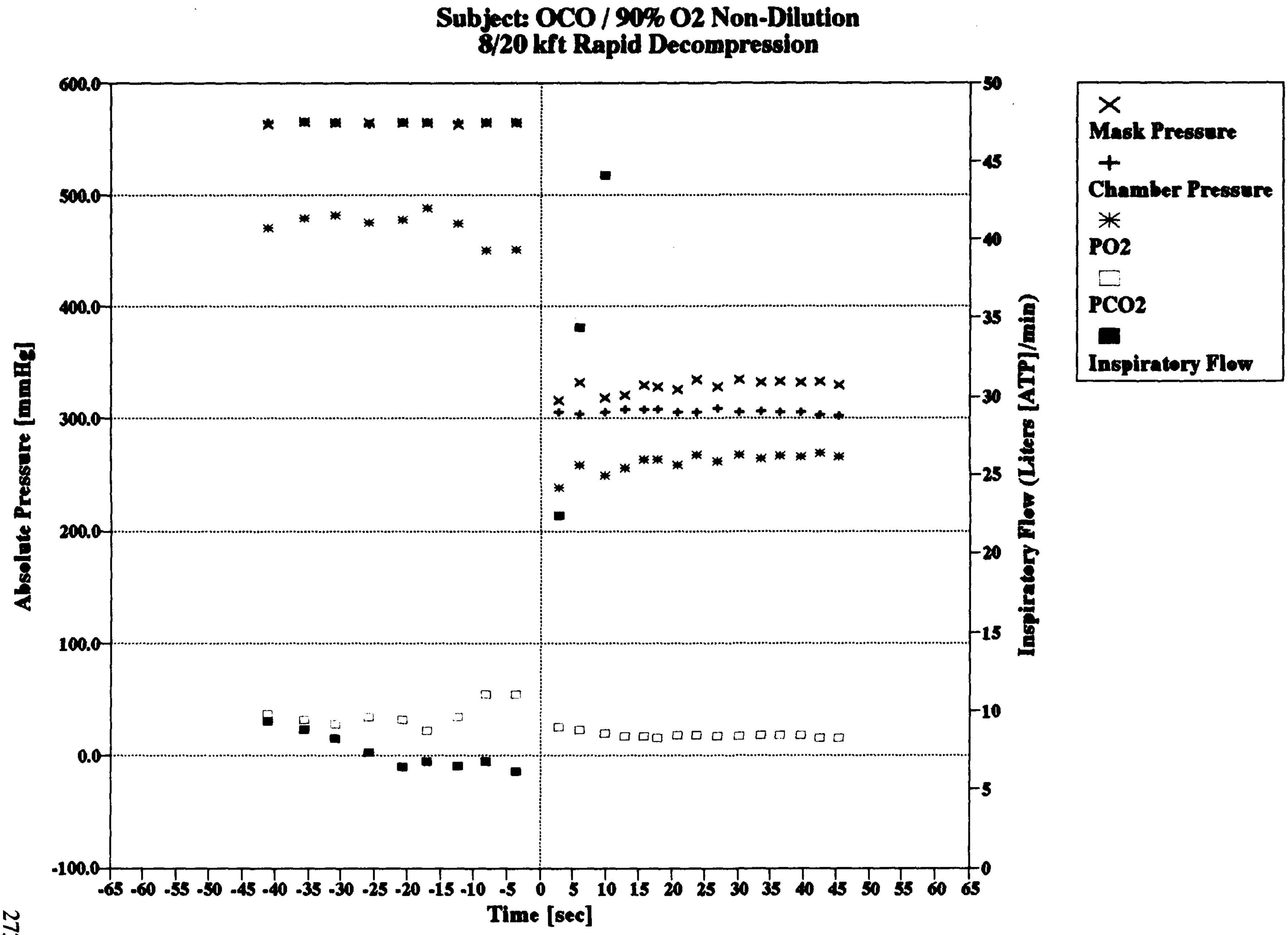
Subject: OCO / 100% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: OCO / 93% O2 Non-Dilution
20/50 kft Rapid Decompression

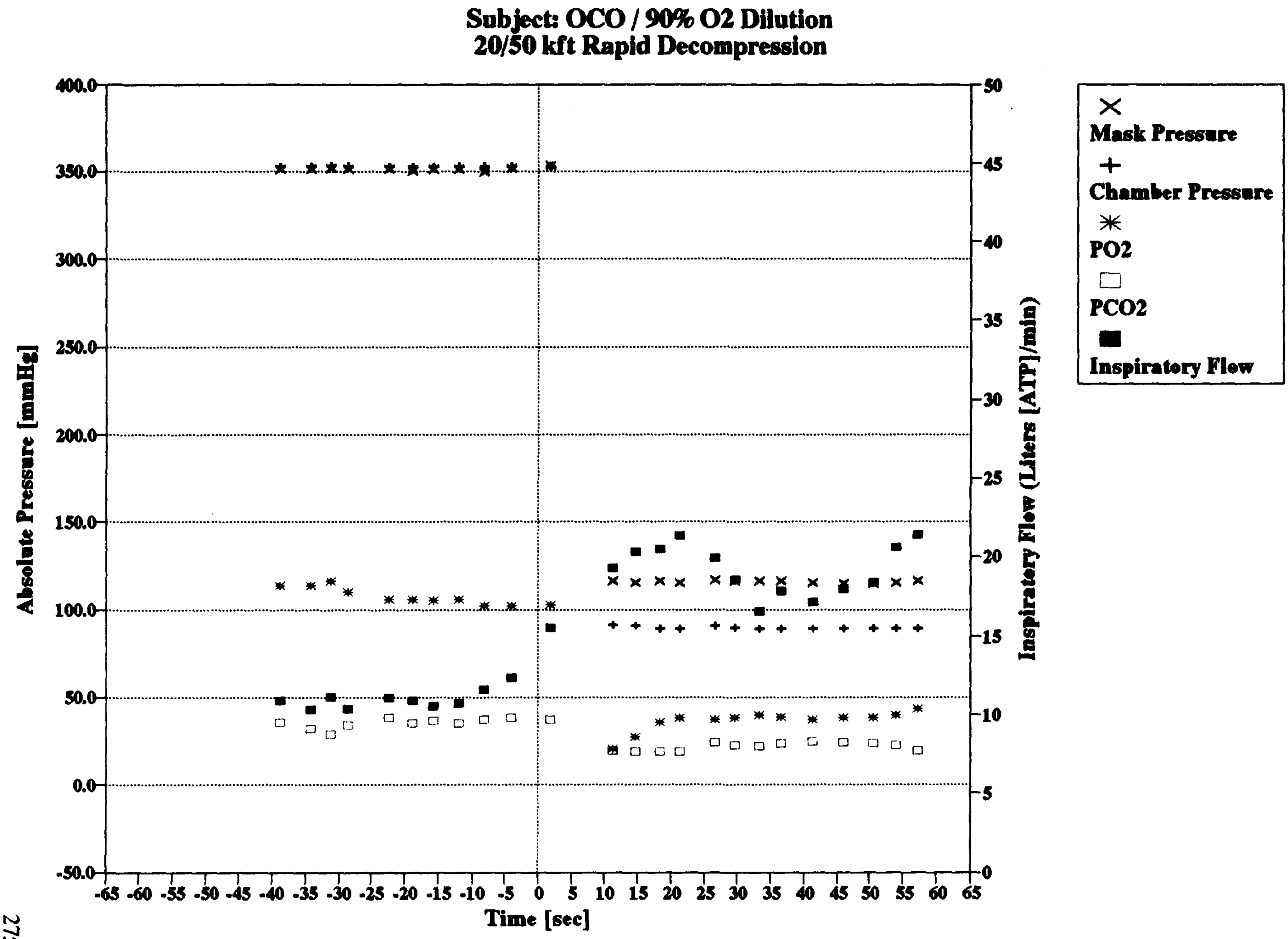


Subject: OCO / 93% O2 Dilution
20/50 kft Rapid Decompression

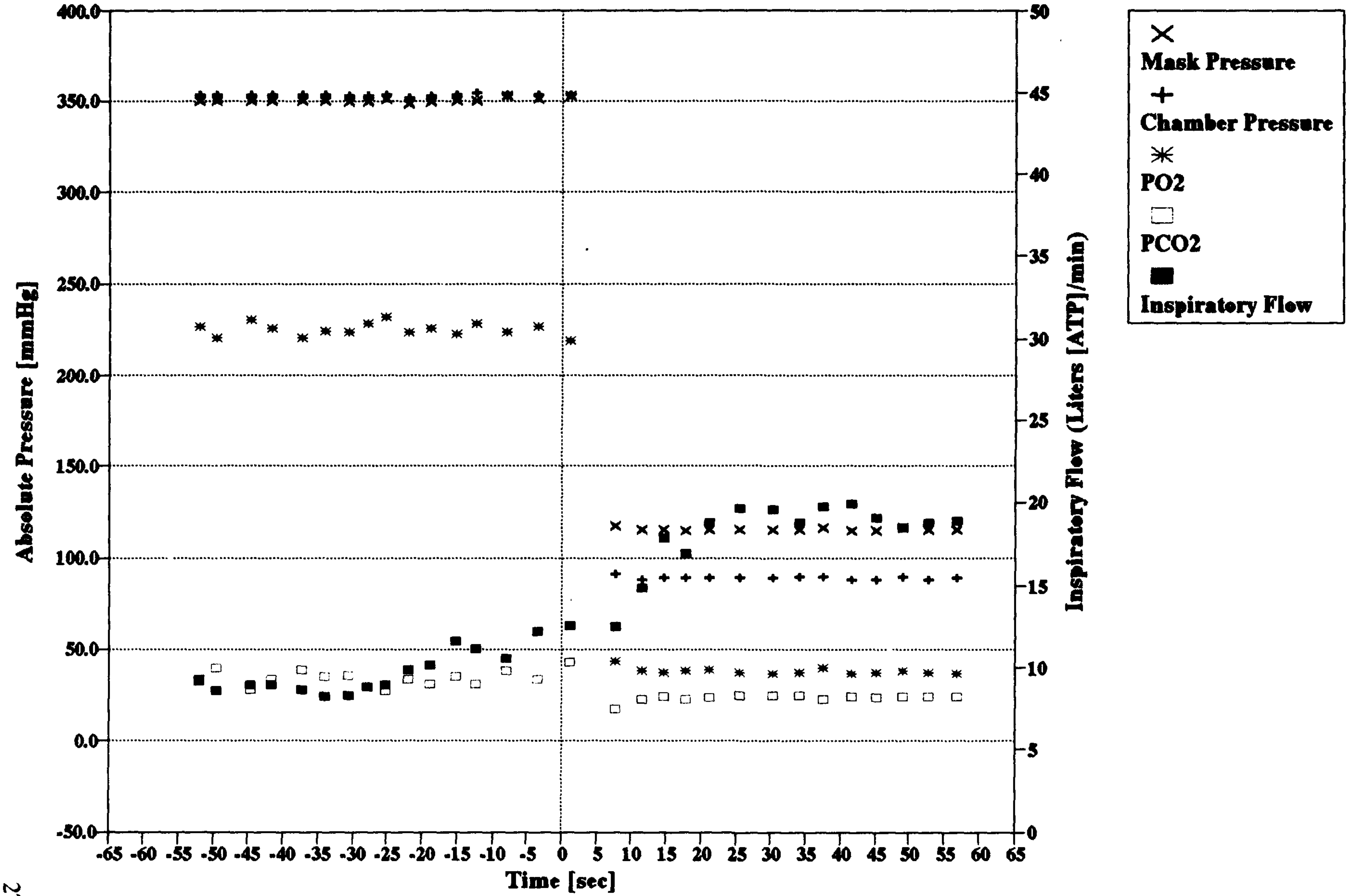




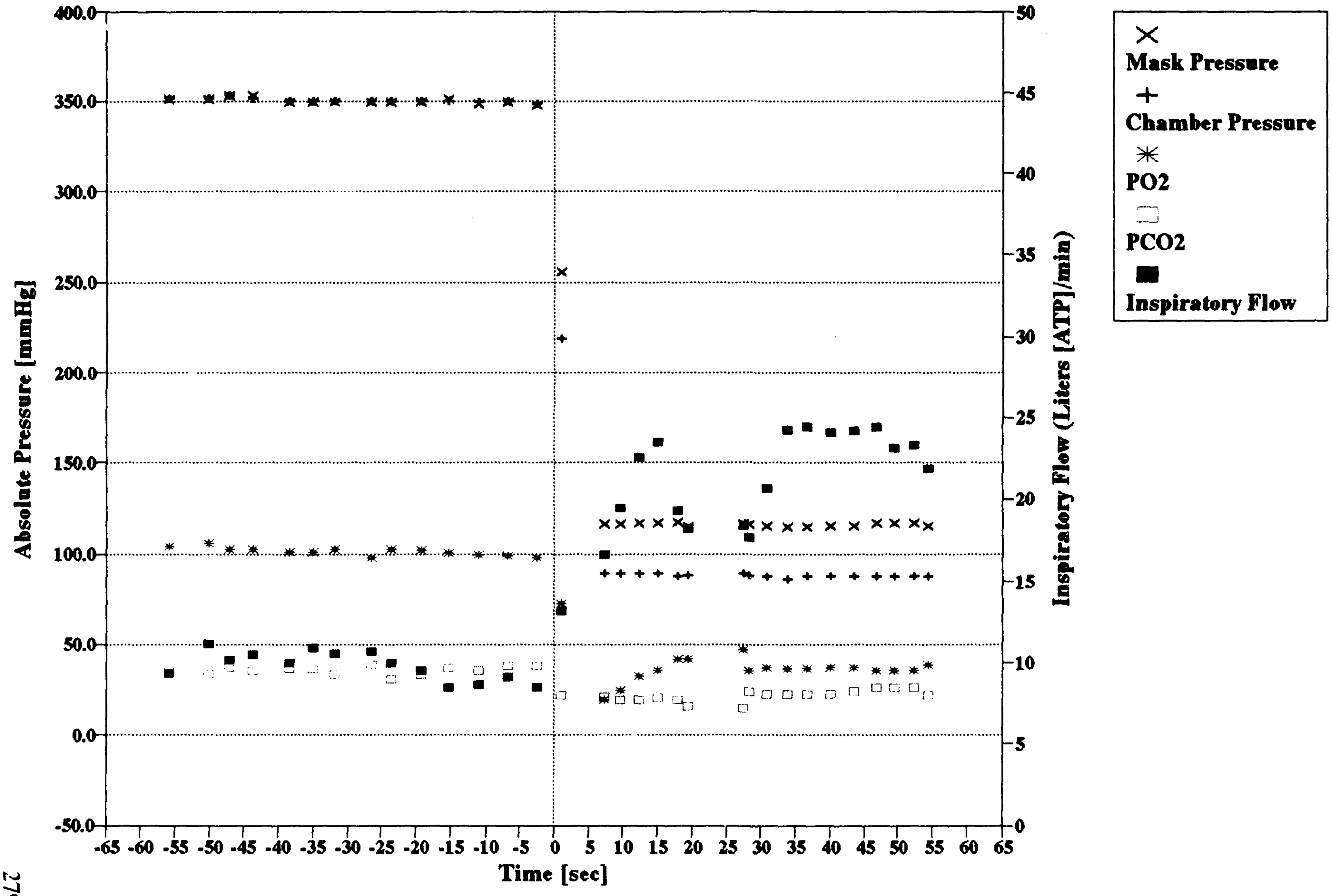
Subject: OCO / 90% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow


Subject: OCO / 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

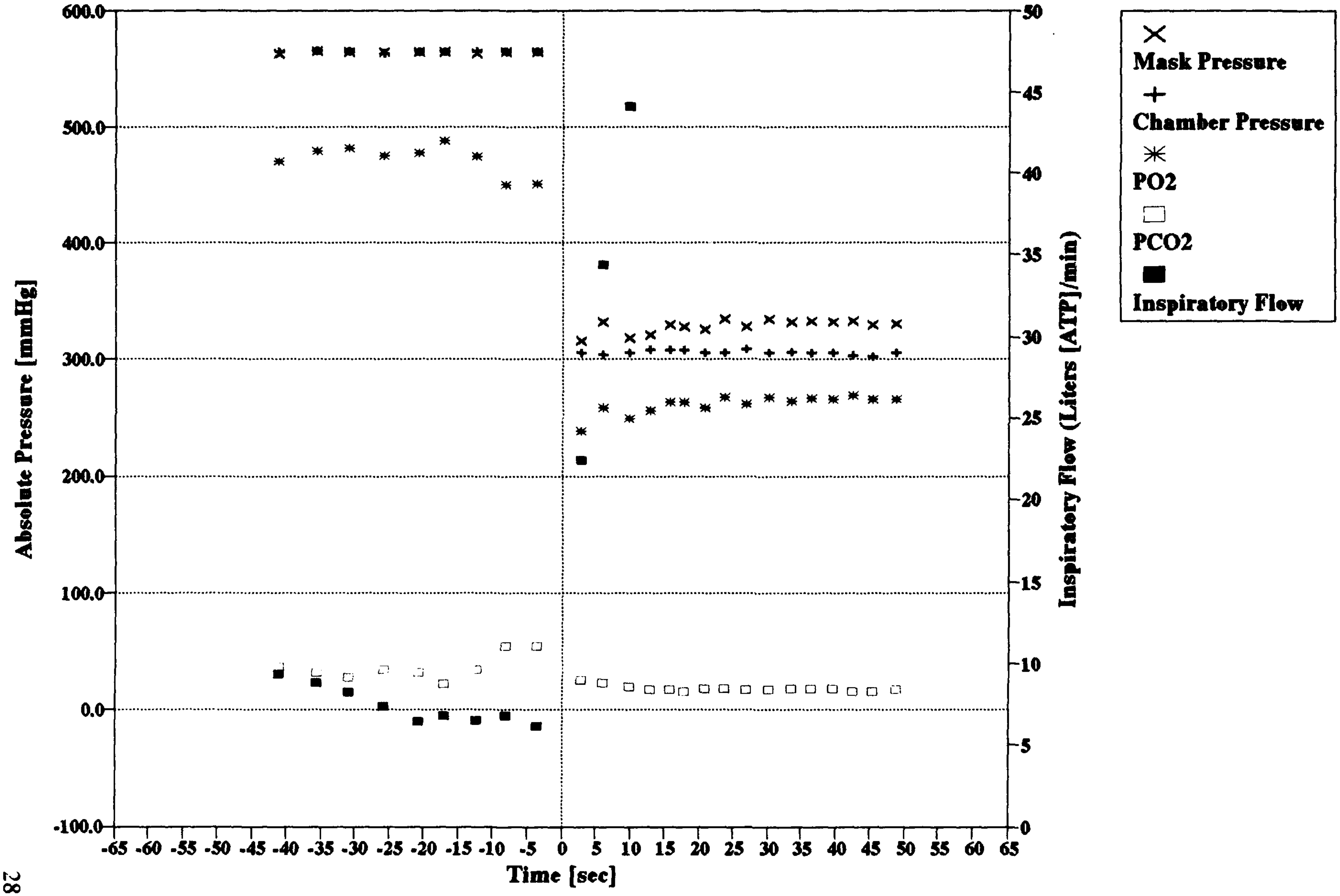
Subject: OCO / 85% O2 Non-Dilution
20/50 kft Rapid Decompression

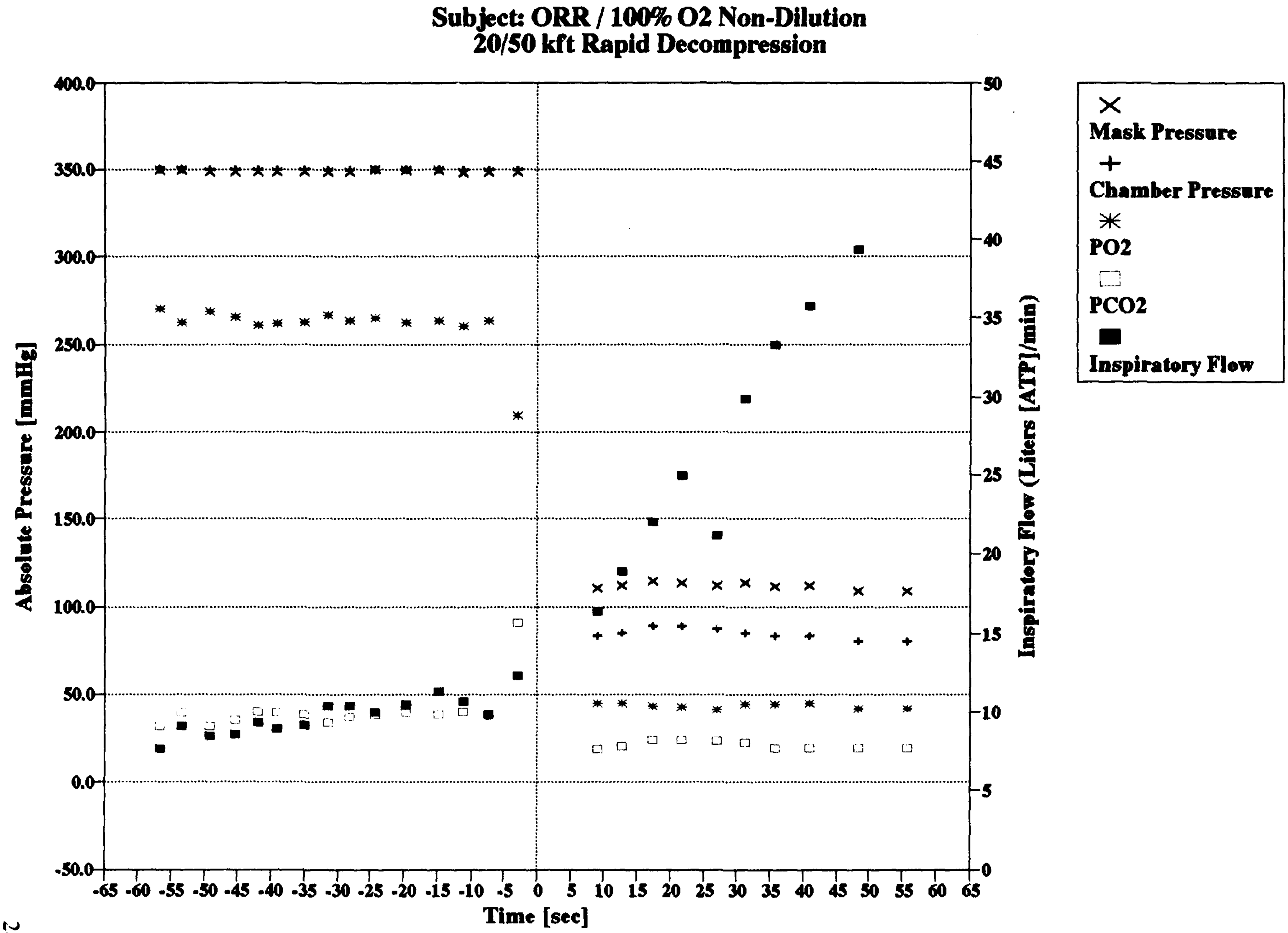


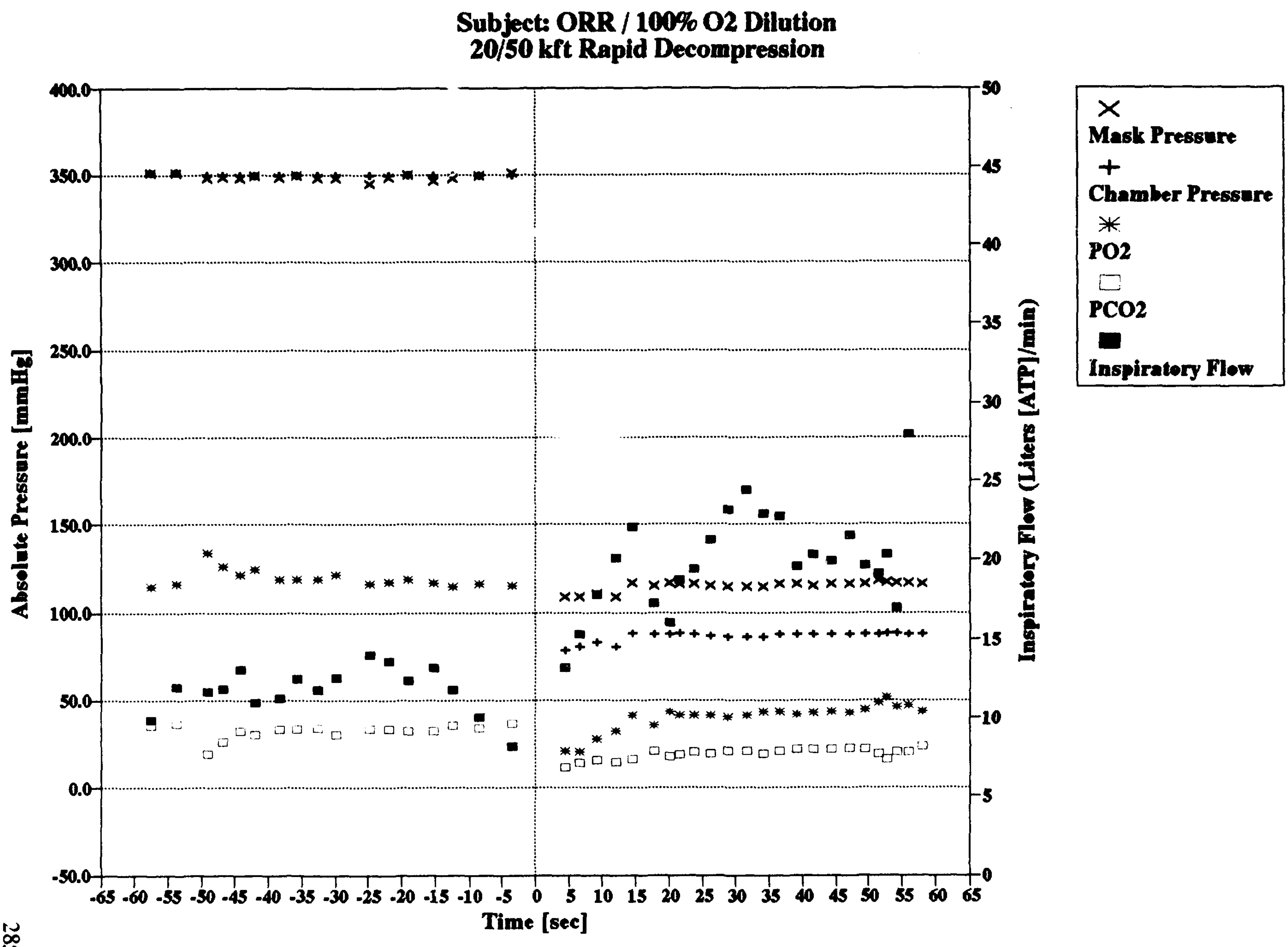
Subject: OCO / 85% O2 Dilution
20/50 kft Rapid Decompression



Subject: OCO / 100% O2 Non-Dilution
8/20 kft Rapid Decompression

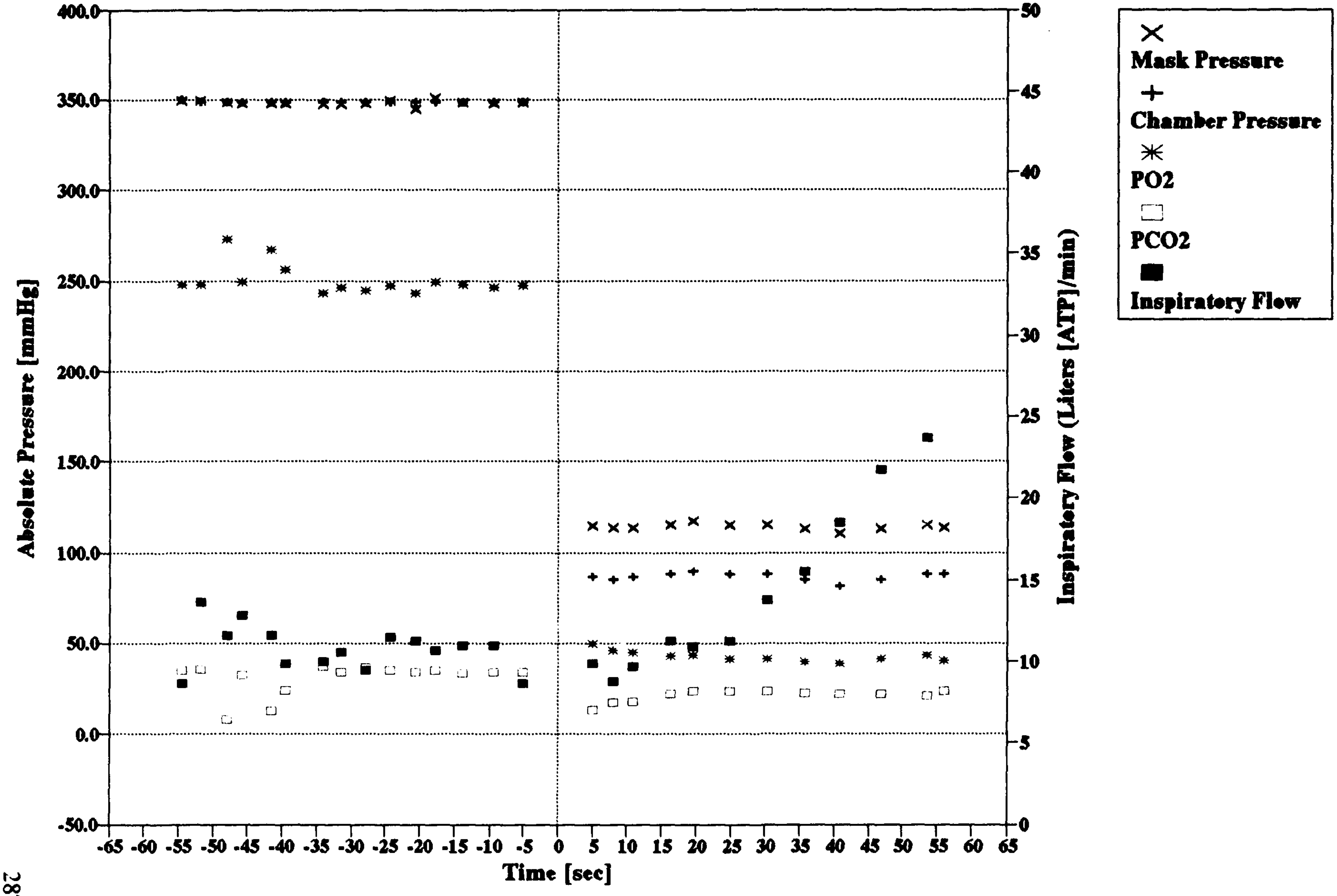



Subject: ORR / 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

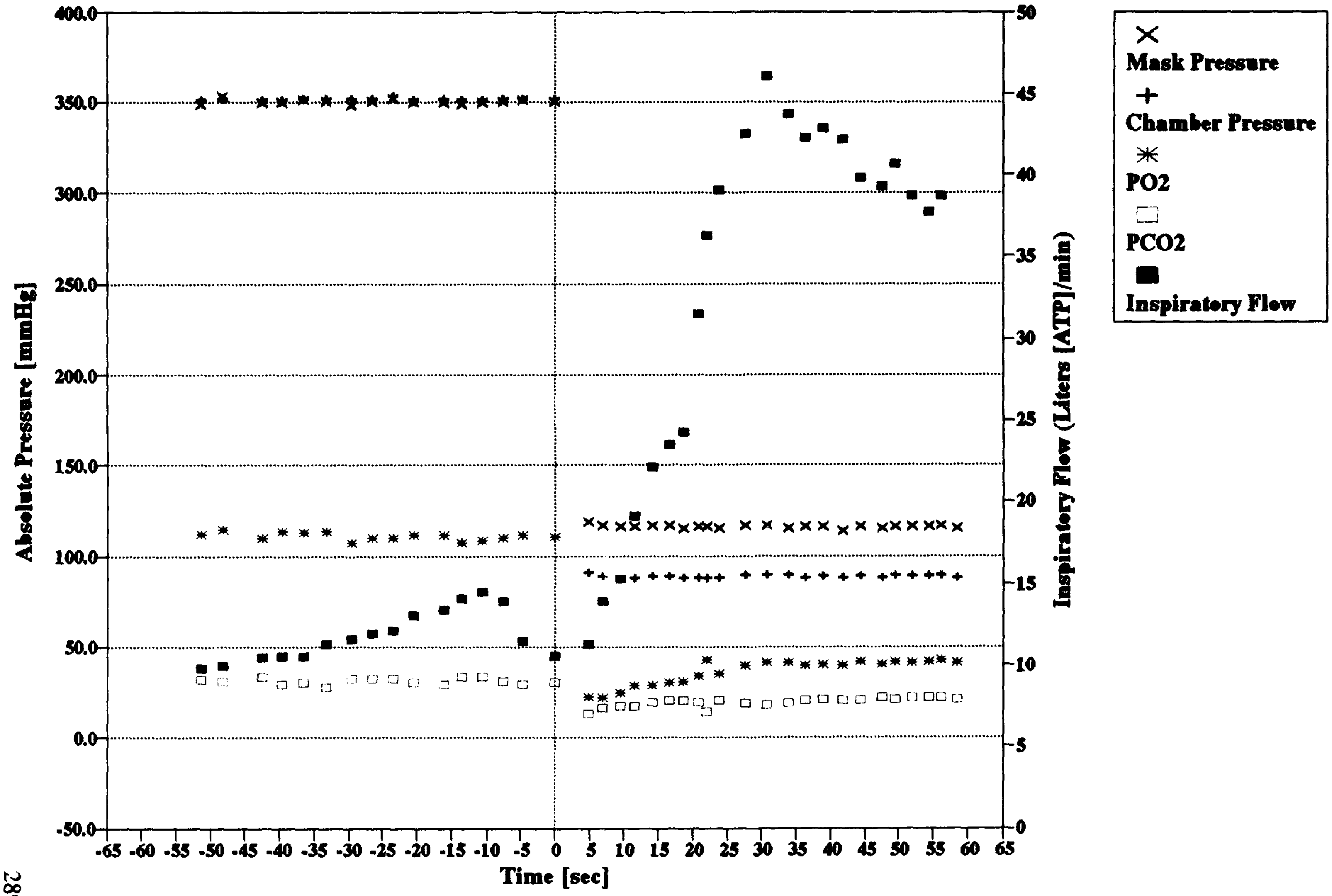


Subject: ORR / 100% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

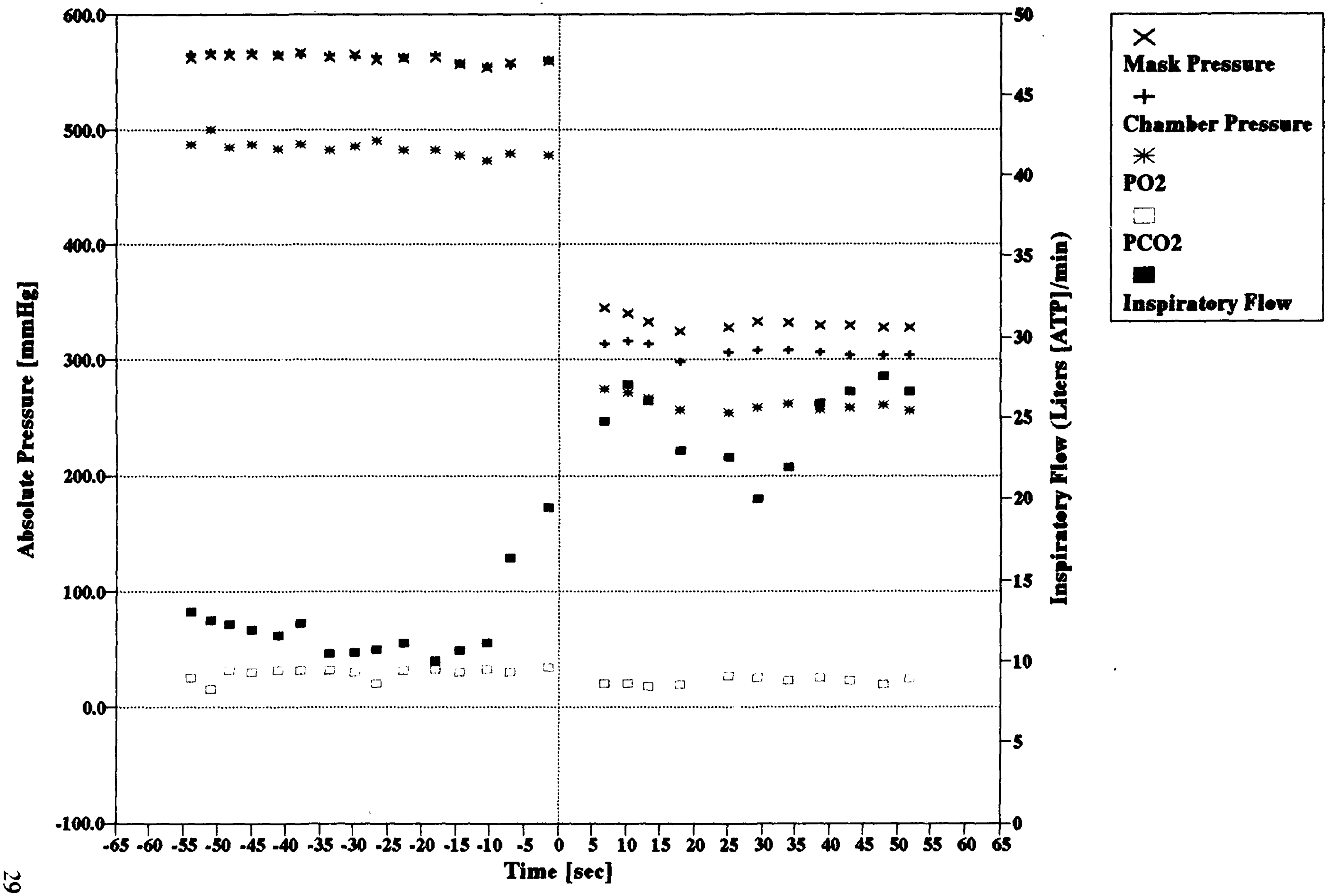
Subject: ORR / 93% O2 Non-Dilution
20/50 kft Rapid Decompression



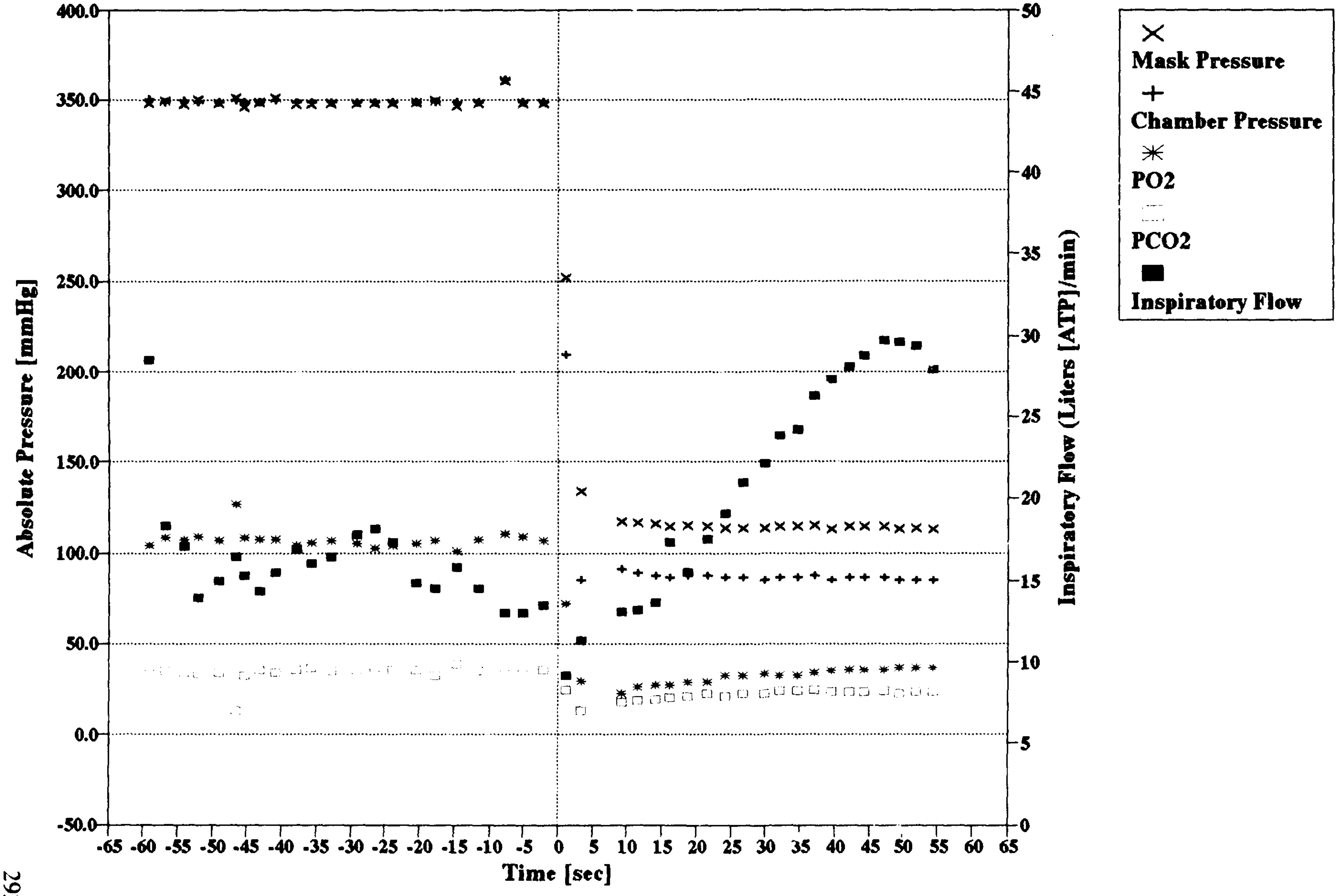
Subject: ORR / 93% O2 Dilution
20/50 kft Rapid Decompression



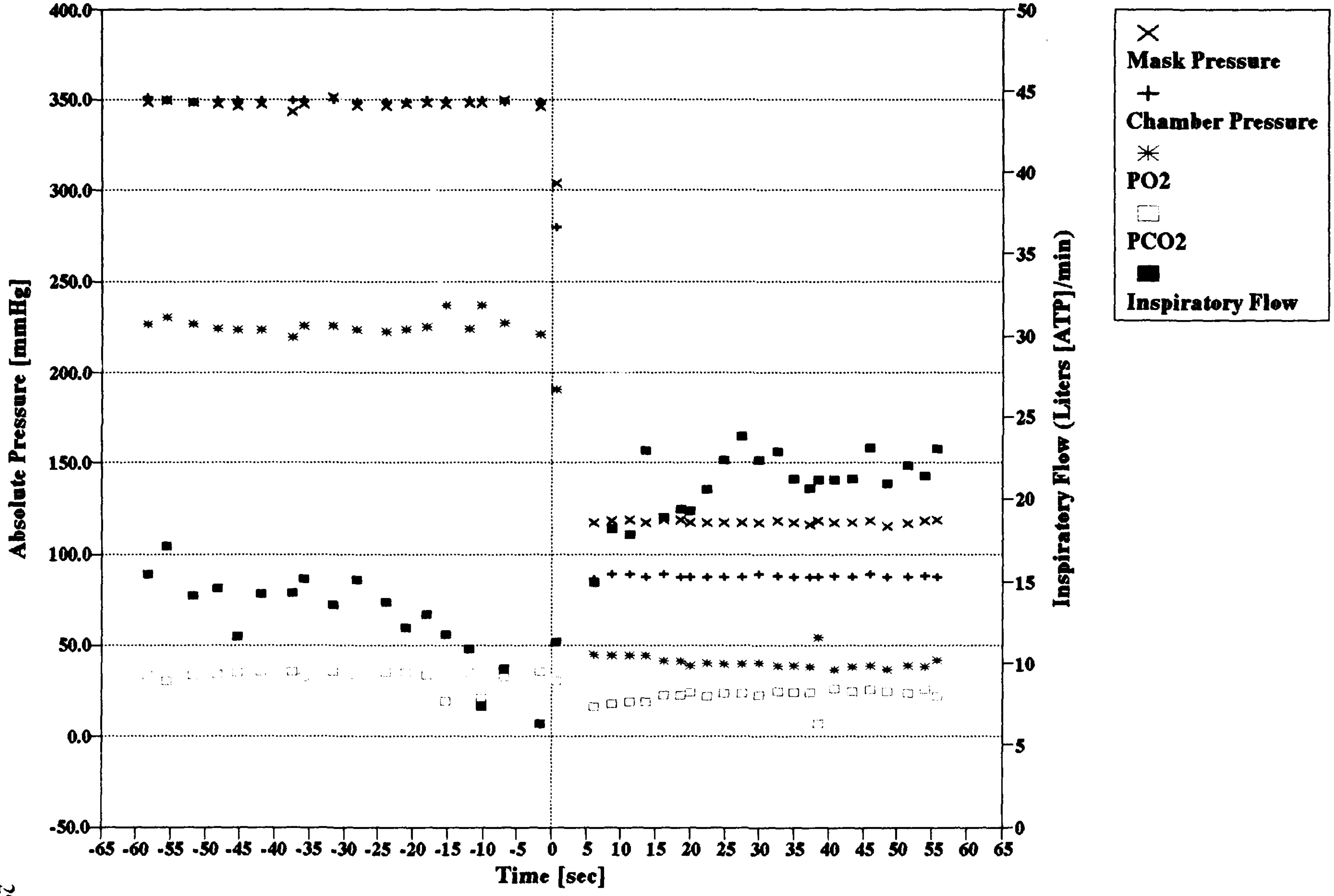
Subject: ORR / 90% O2 Non-Dilution
8/20 kft Rapid Decompression

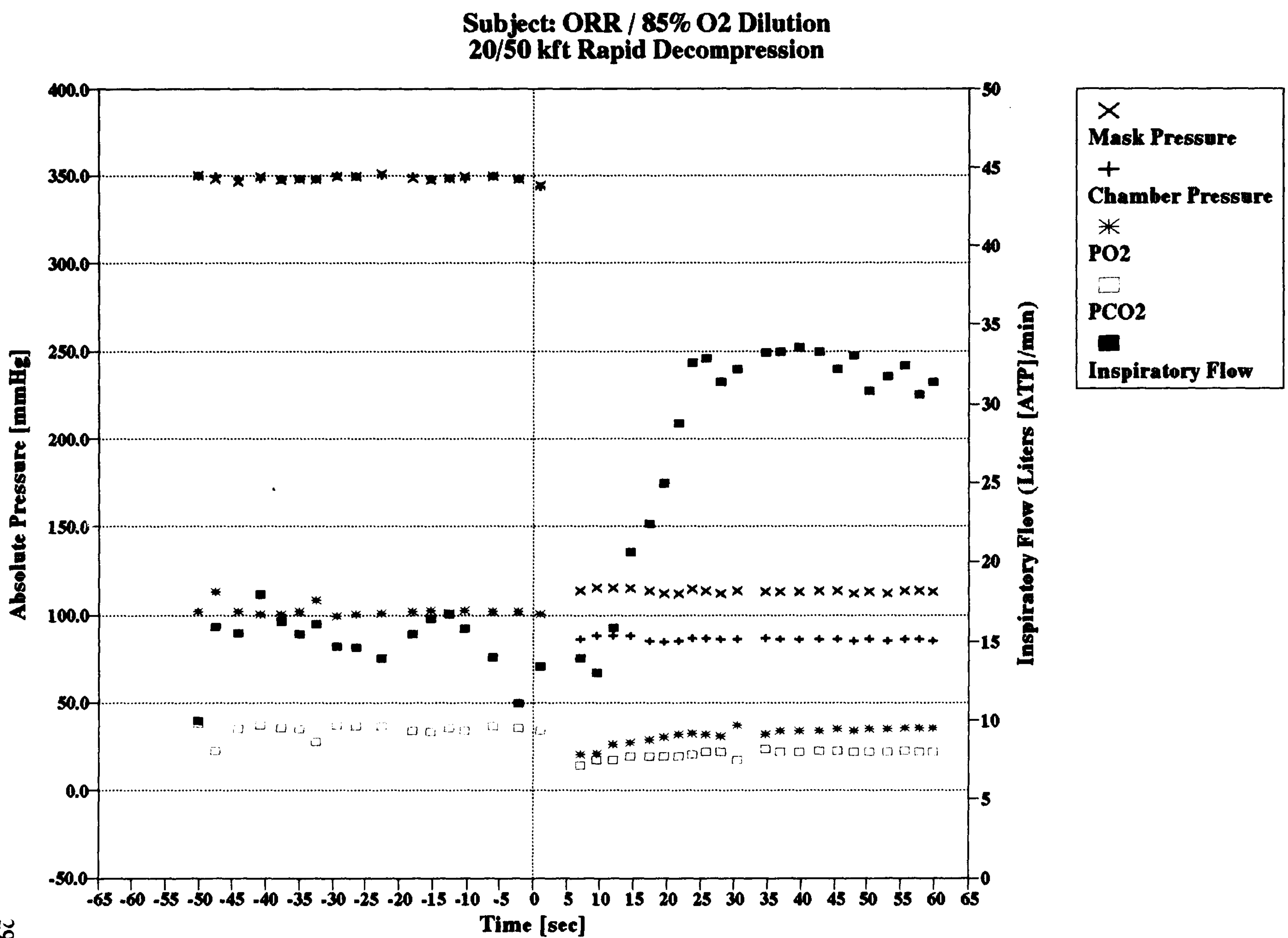


Subject: ORR / 90% O2 Dilution
20/50 kft Rapid Decompression

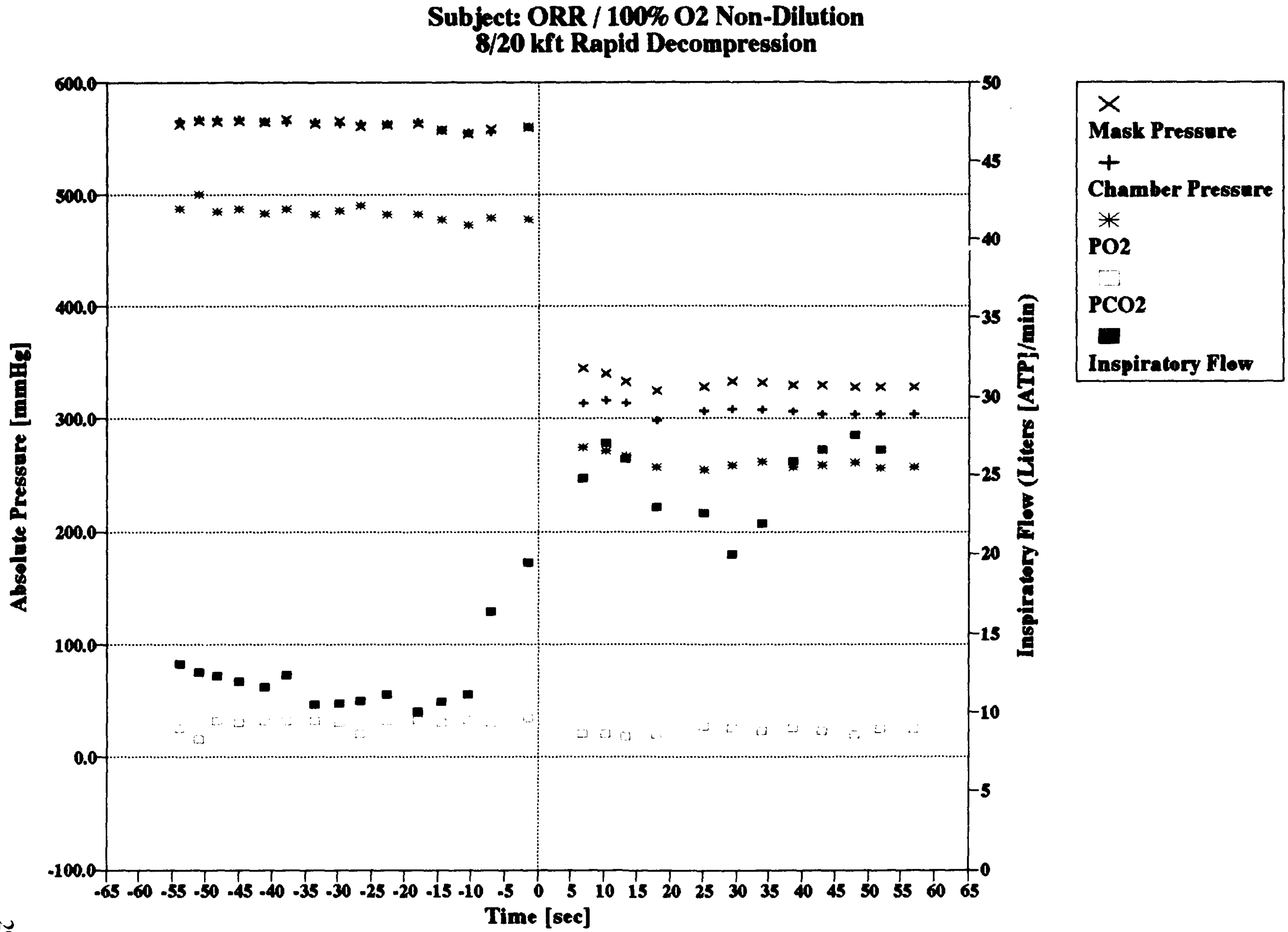


Subject: ORR / 85% O2 Non-Dilution
20/50 kft Rapid Decompression

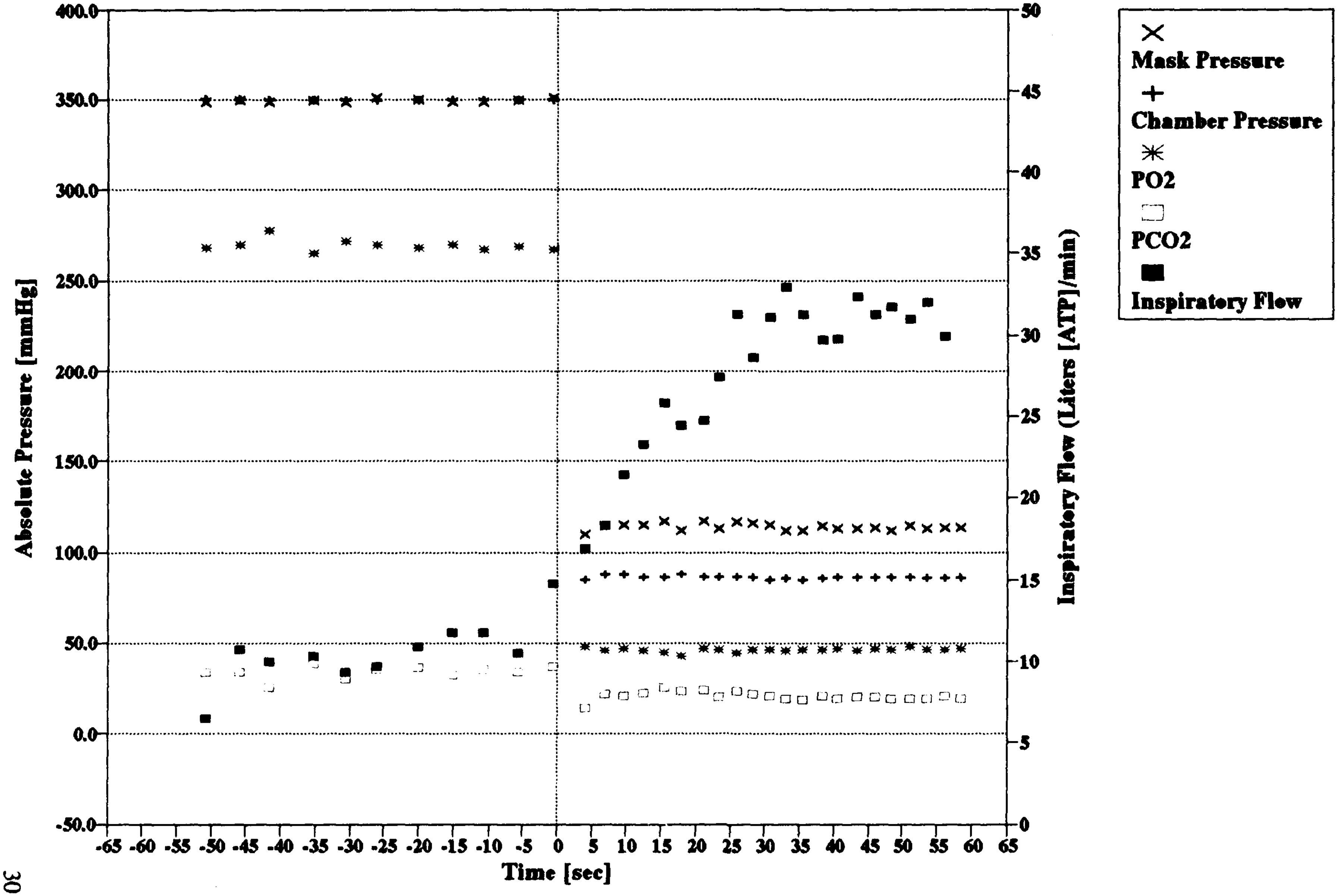




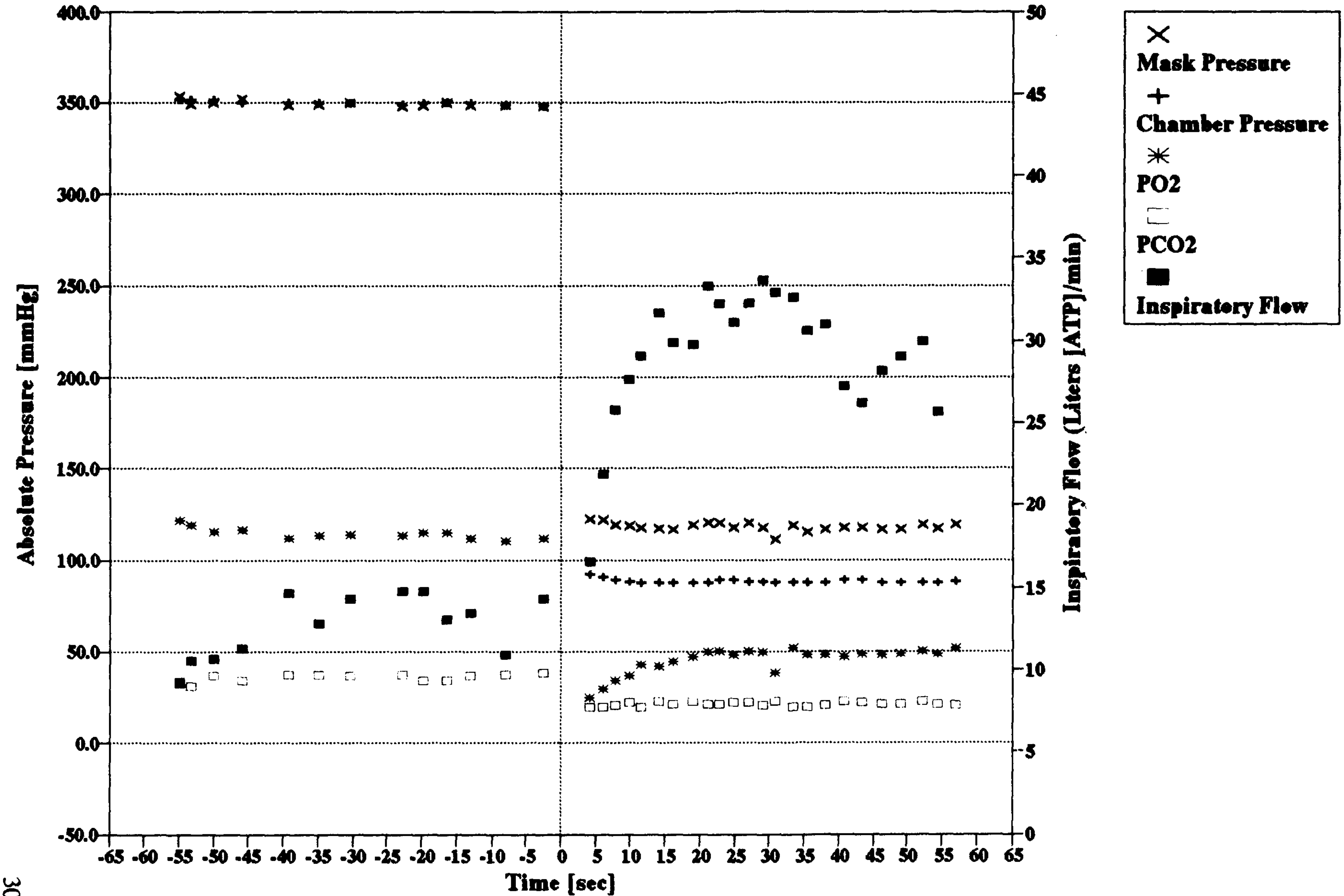
Subject: ORR / 85% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

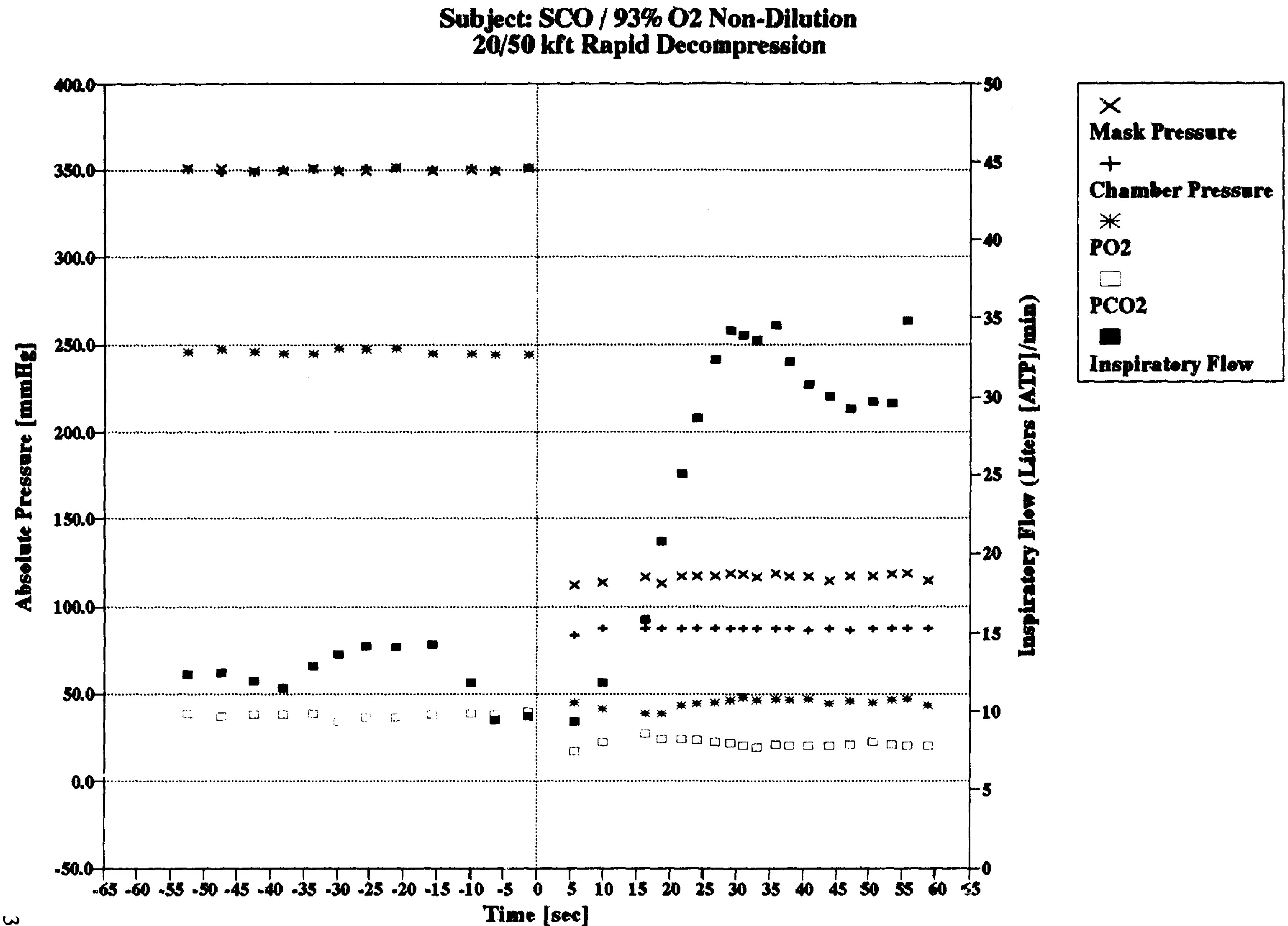

Subject: ORR / 100% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

Subject: SCO / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



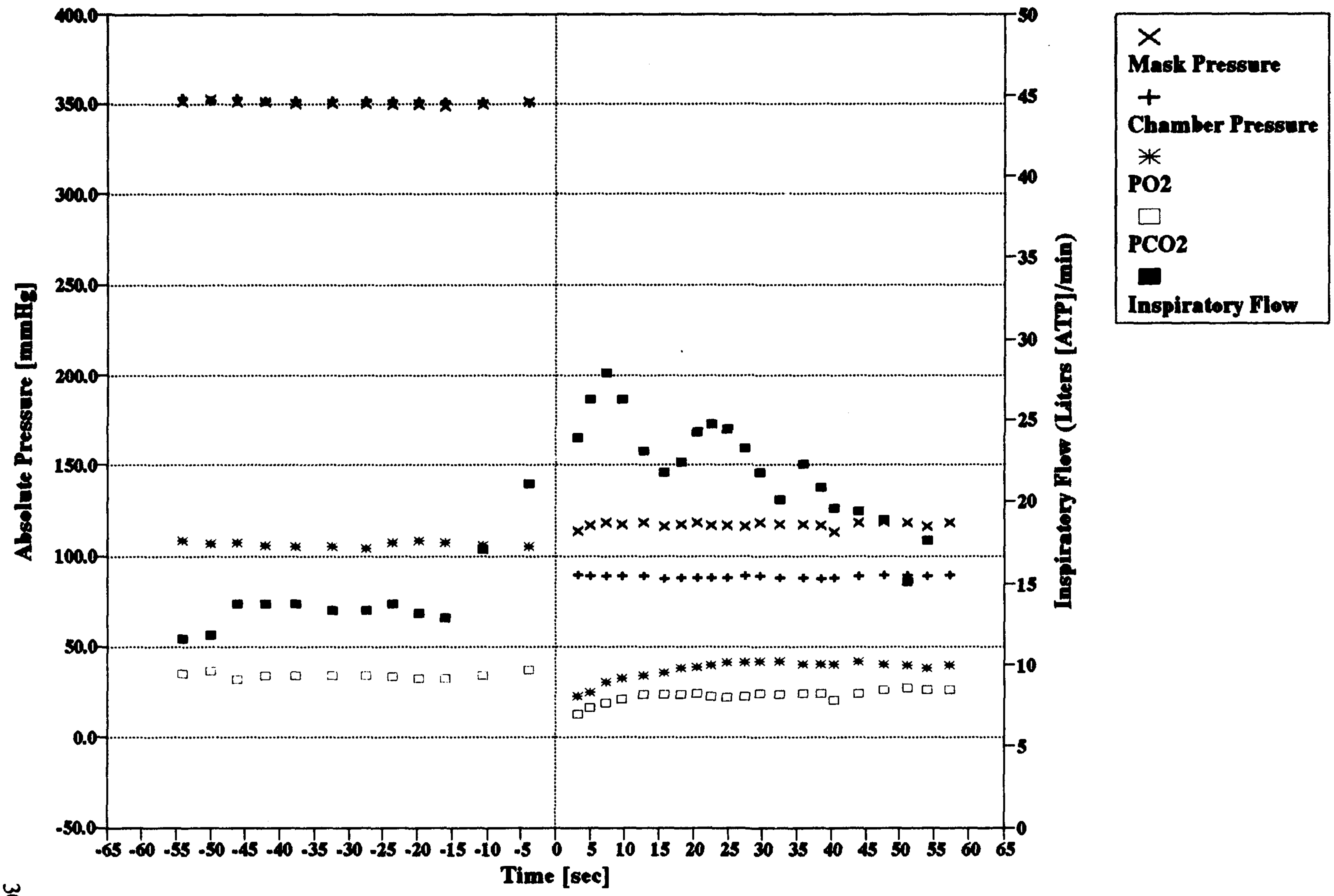
Subject: SCO / 100% O2 Dilution
20/50 kft Rapid Decompression



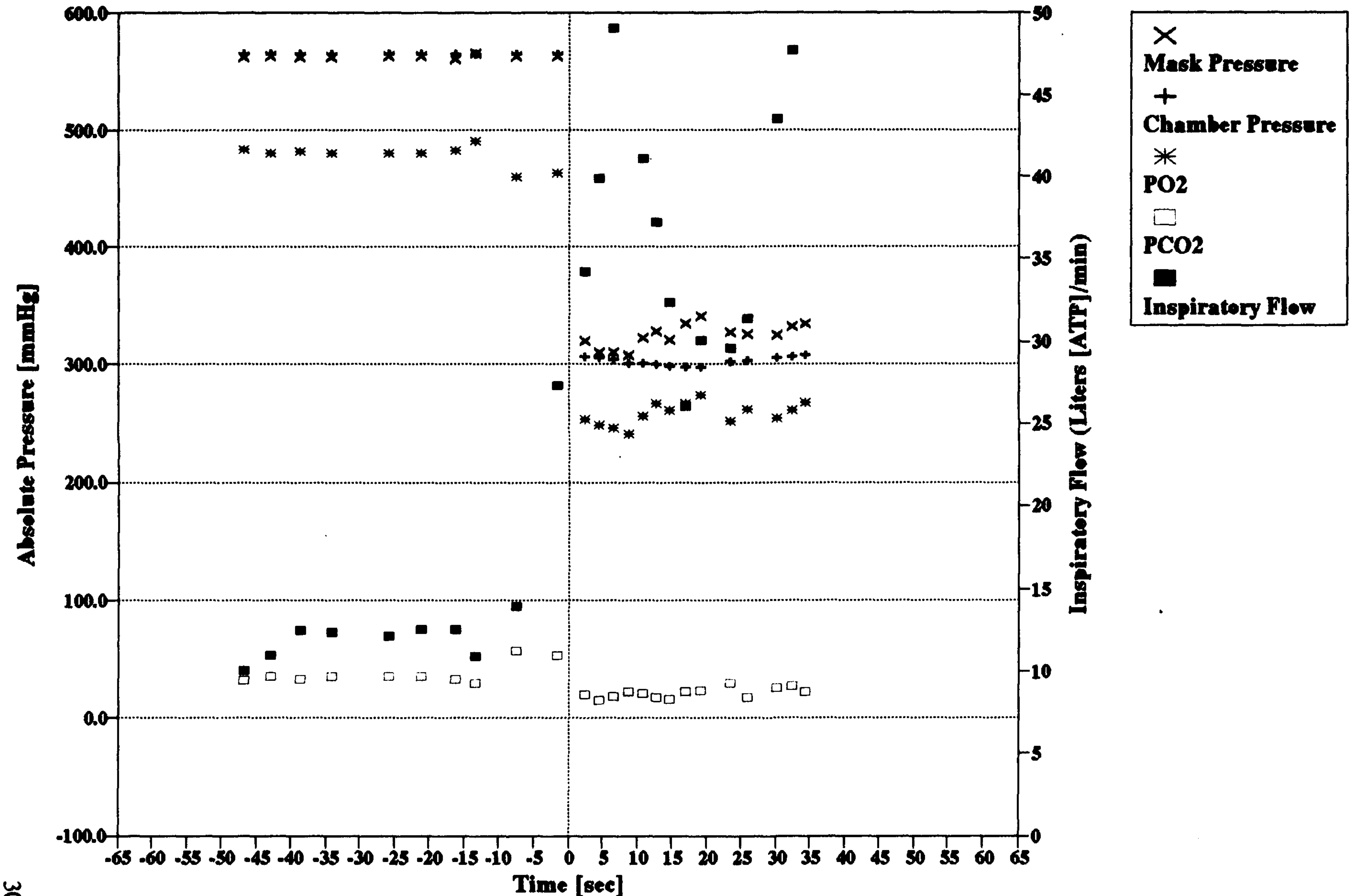


Subject: SCO / 93% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

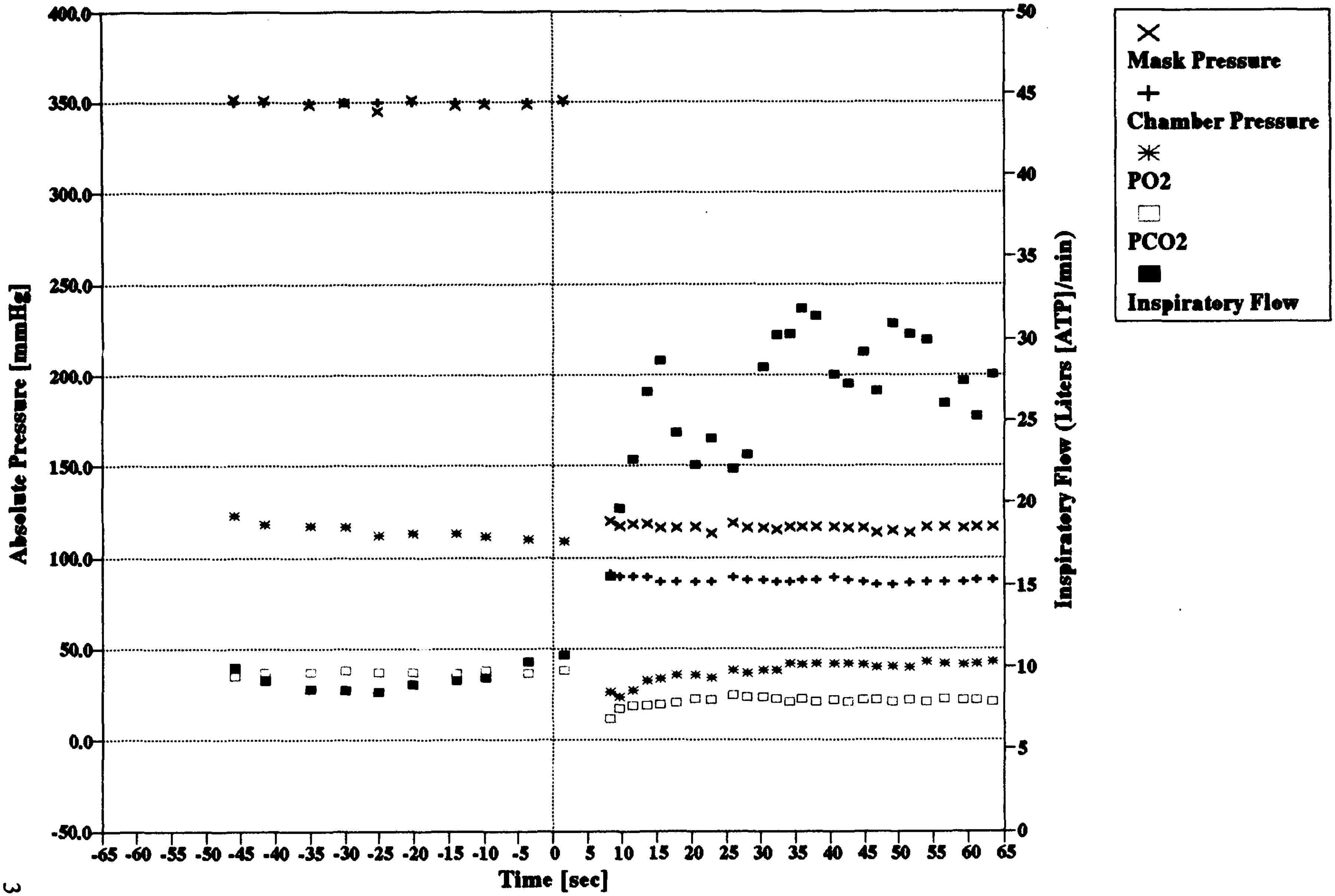
Subject: SCO / 93% O2 Dilution
20/50 kft Rapid Decompression



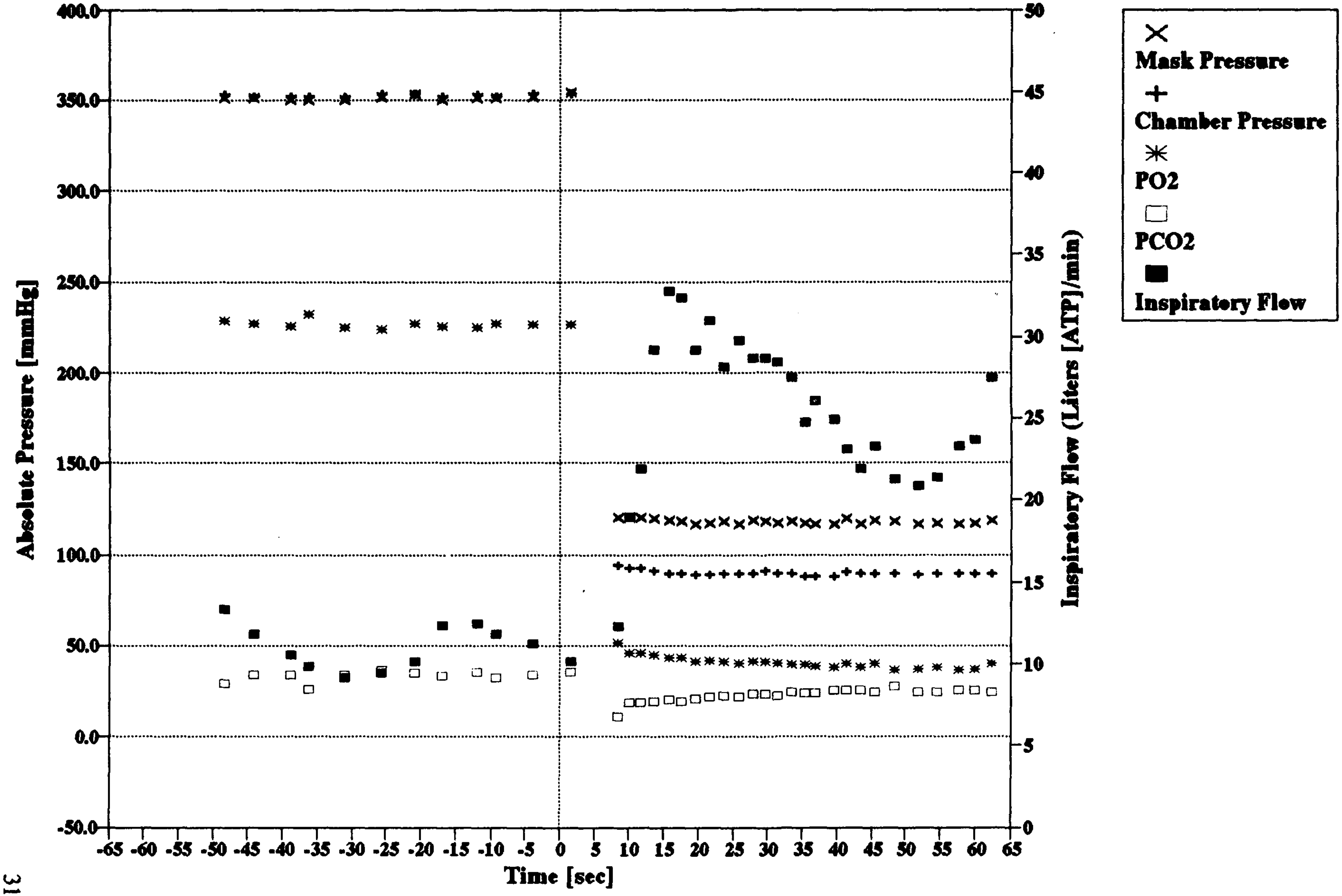
Subject: SCO / 90% O2 Non-Dilution
8/20 kft Rapid Decompression



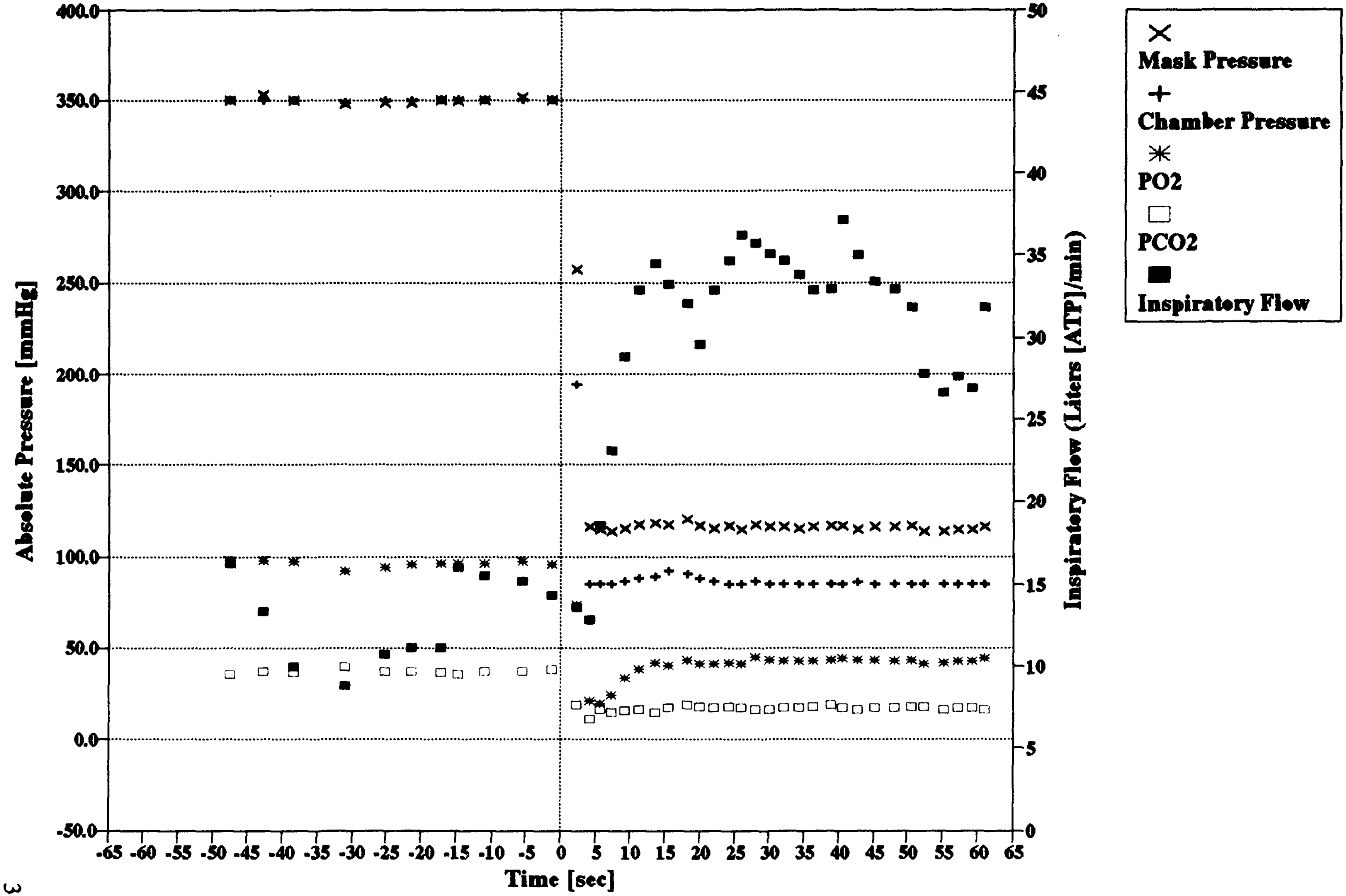
Subject: SCO / 90% O2 Dilution
20/50 kft Rapid Decompression



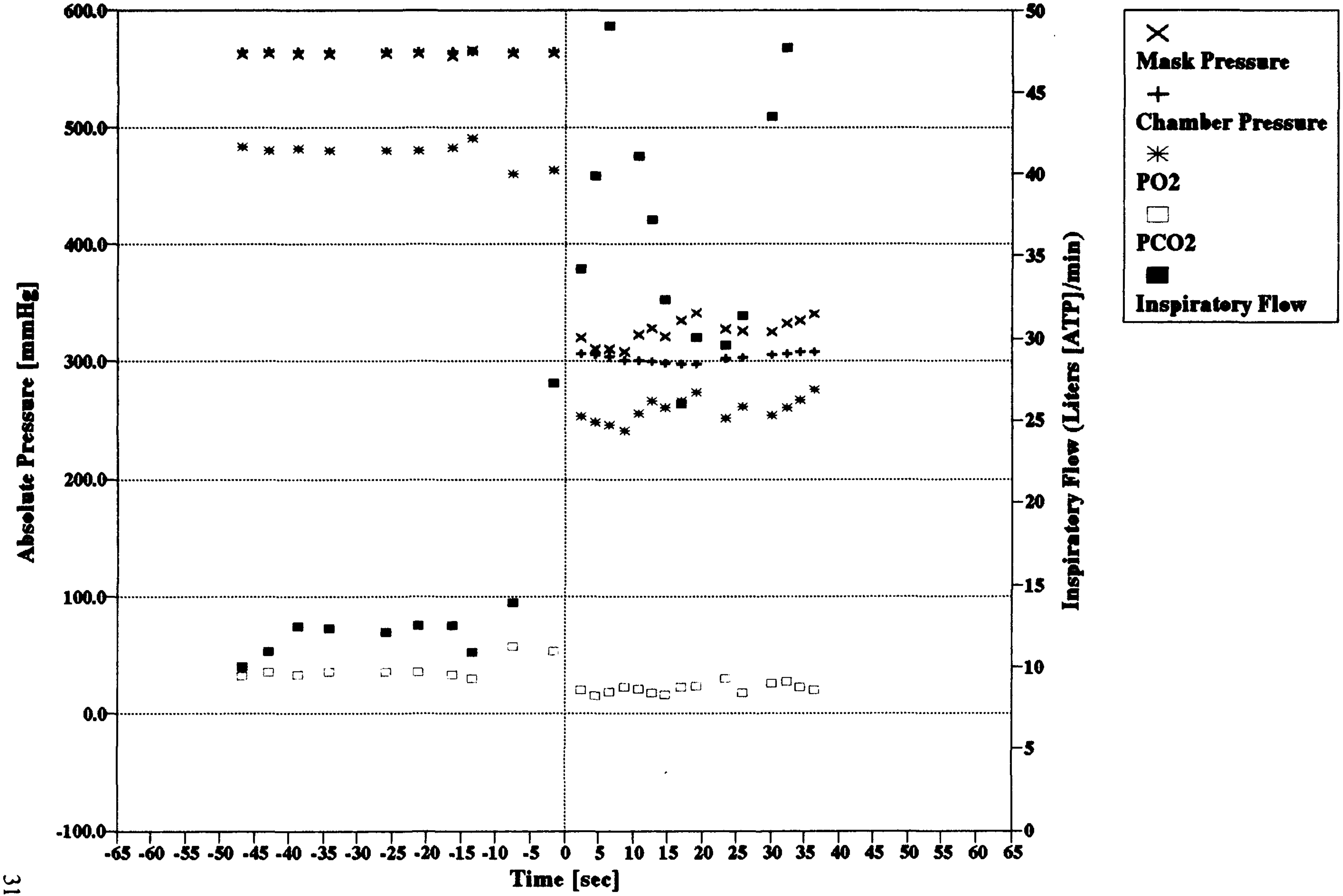
Subject: SCO / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



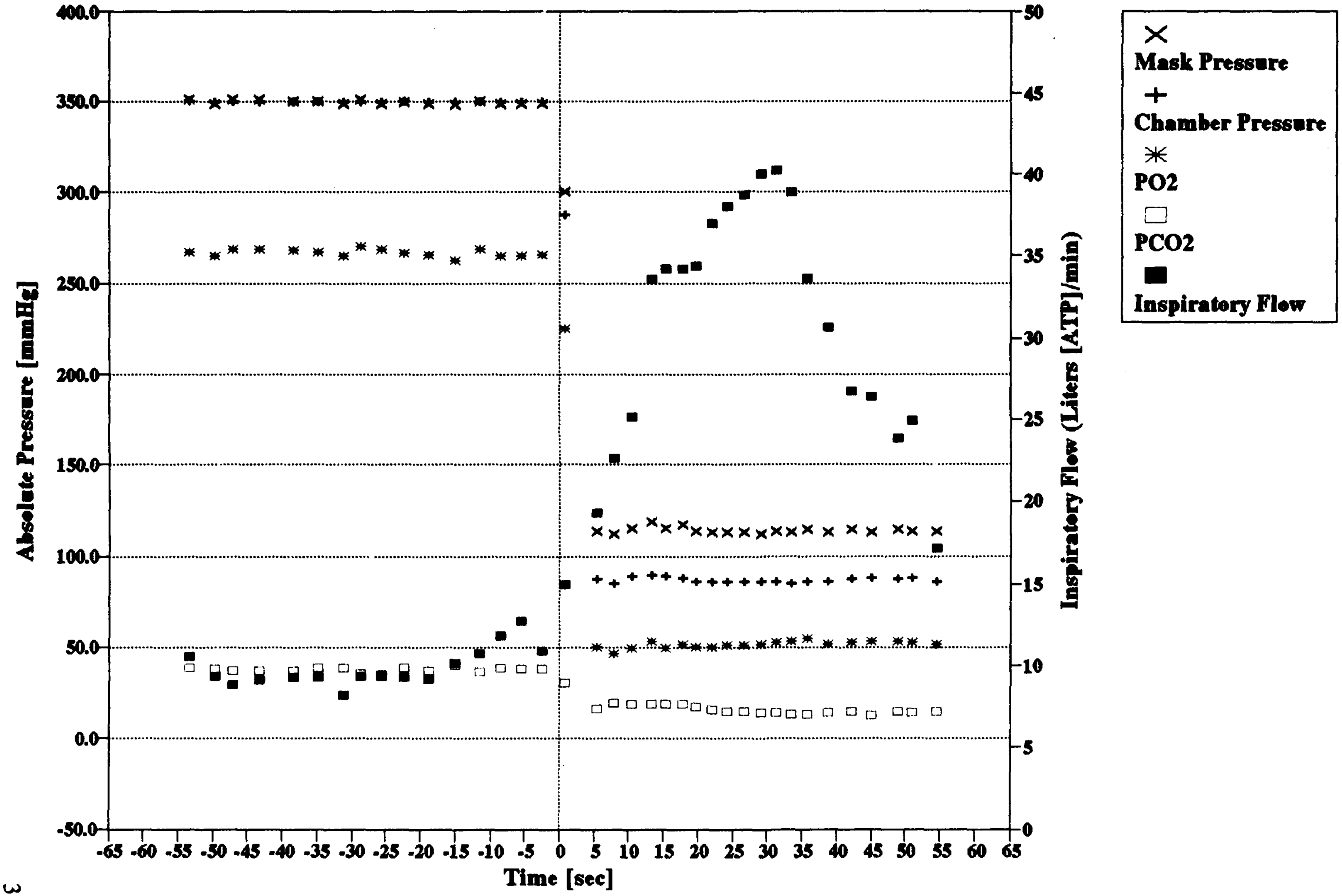
Subject: SCO / 85% O2 Dilution
20/50 kft Rapid Decompression



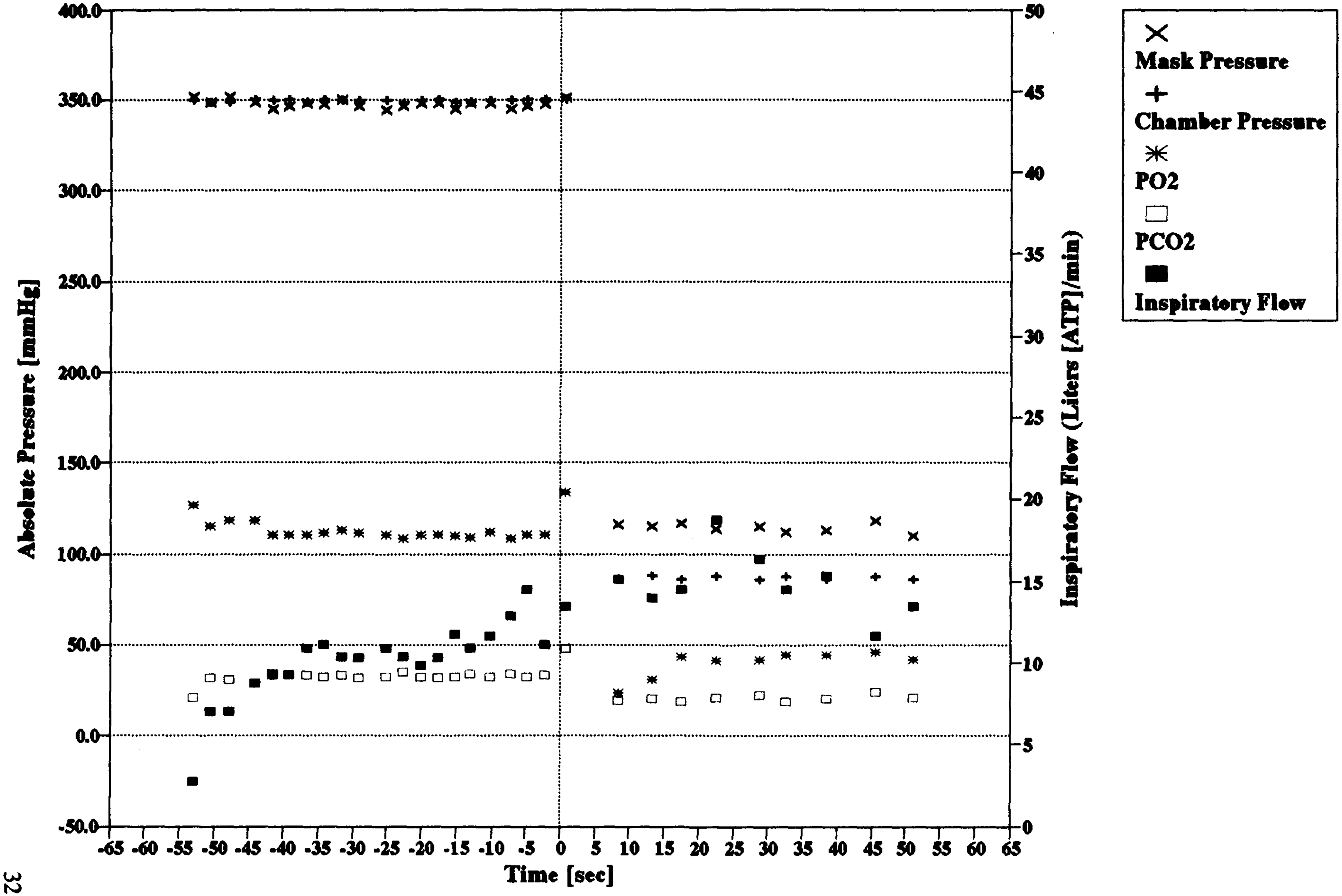
Subject: SCO / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



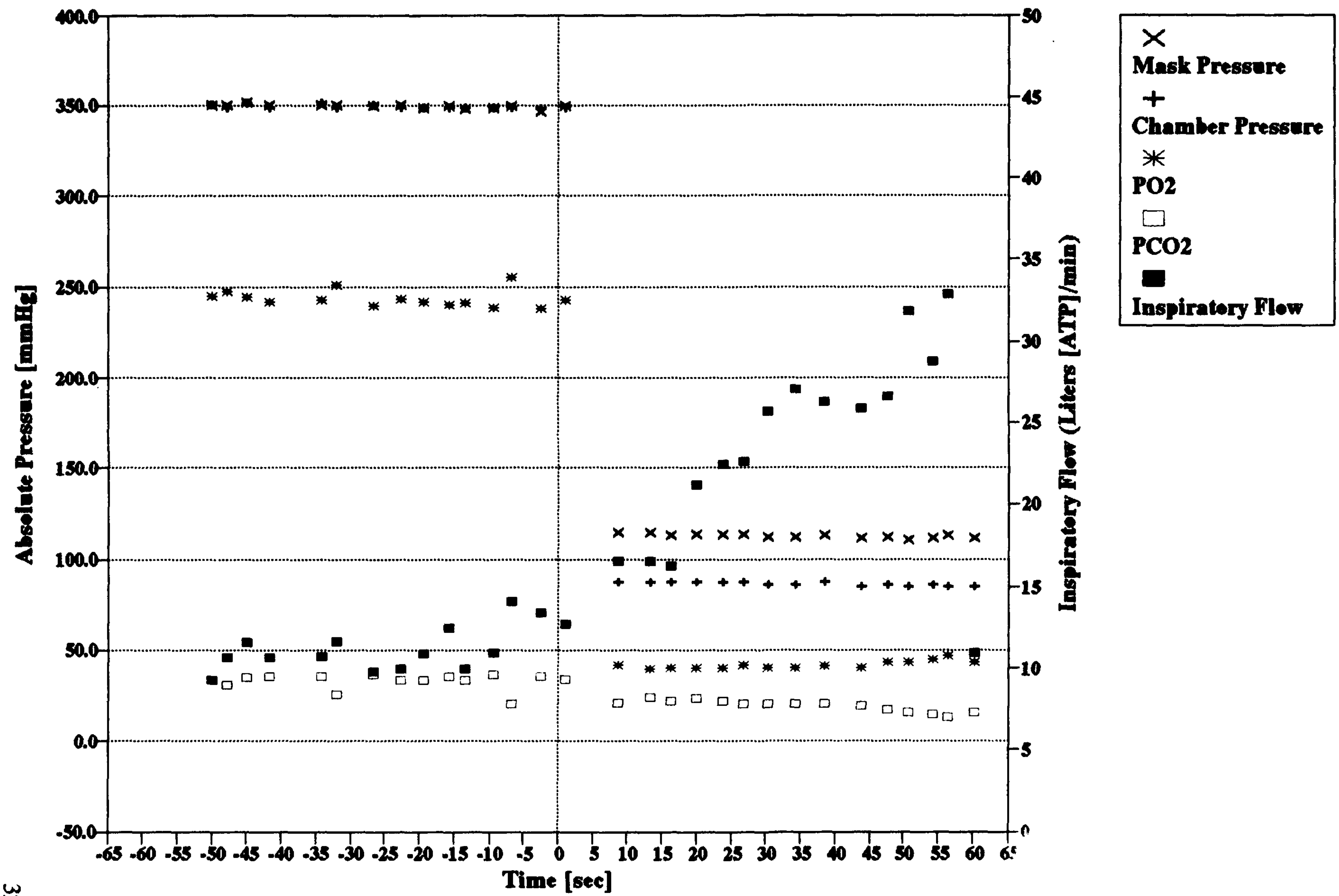
Subject: SHI / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



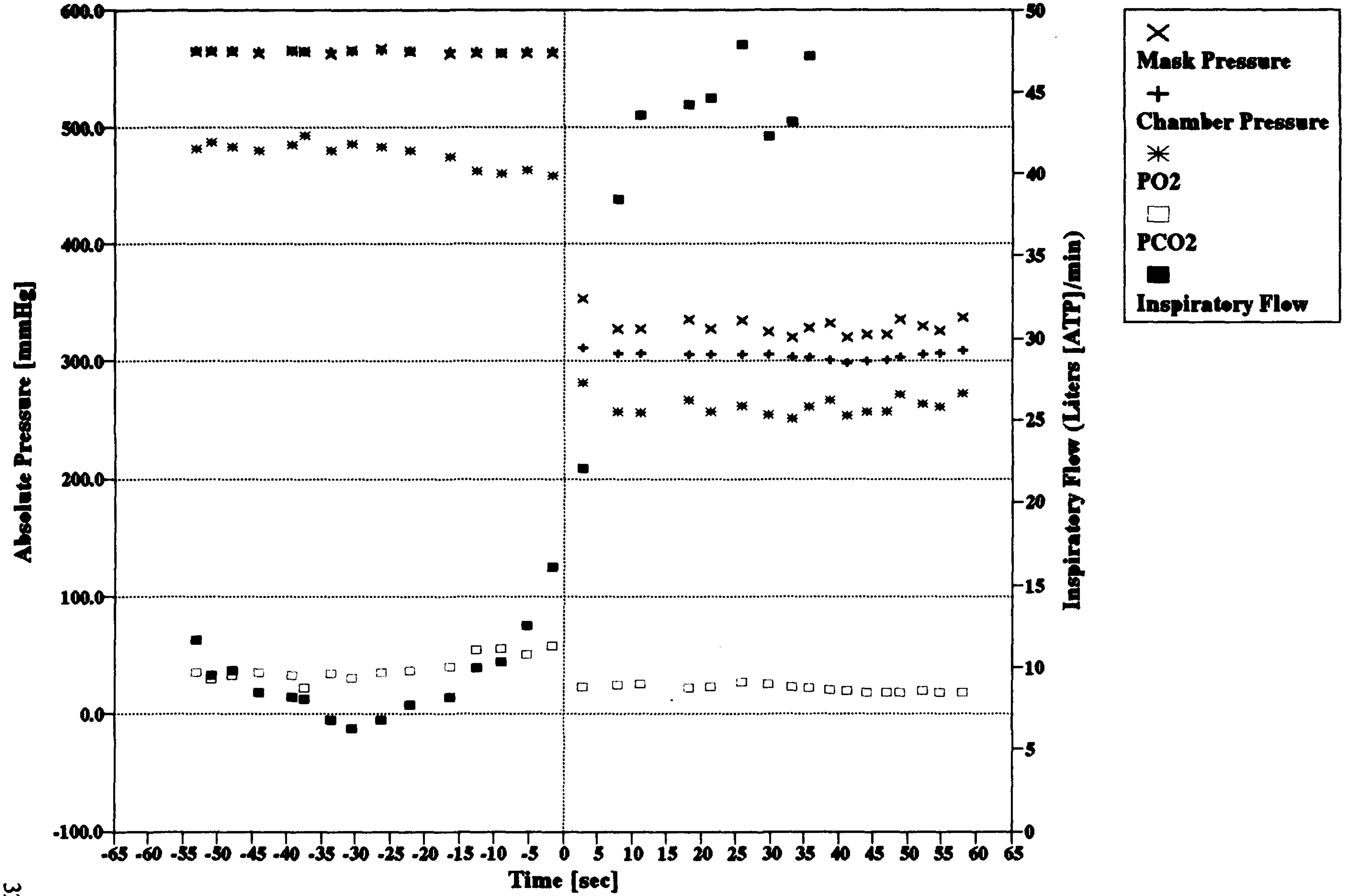
Subject: SHI / 100% O2 Dilution
20/50 kft Rapid Decompression



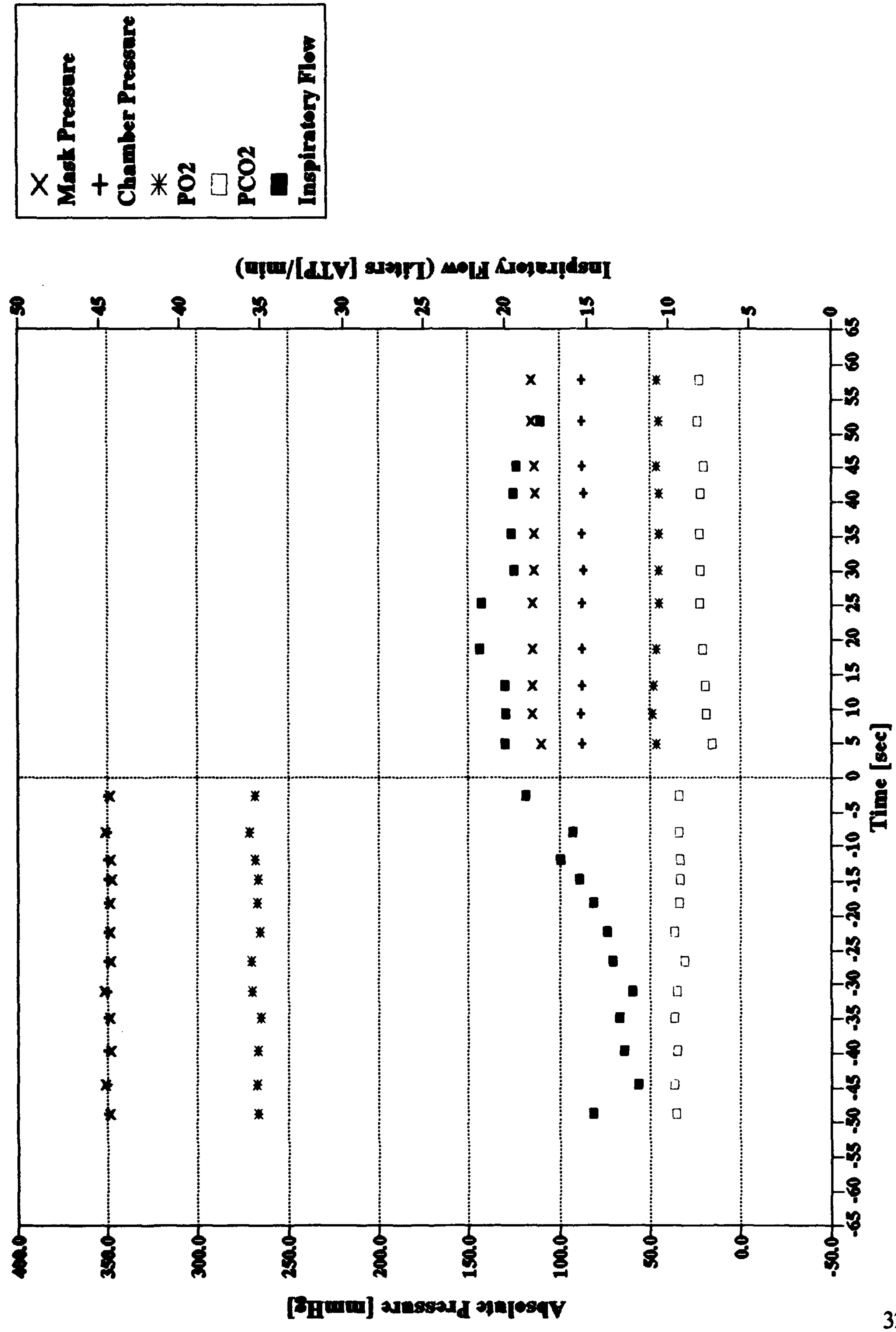
Subject: SHI / 90% O2 Non-Dilution
20/50 kft Rapid Decompression



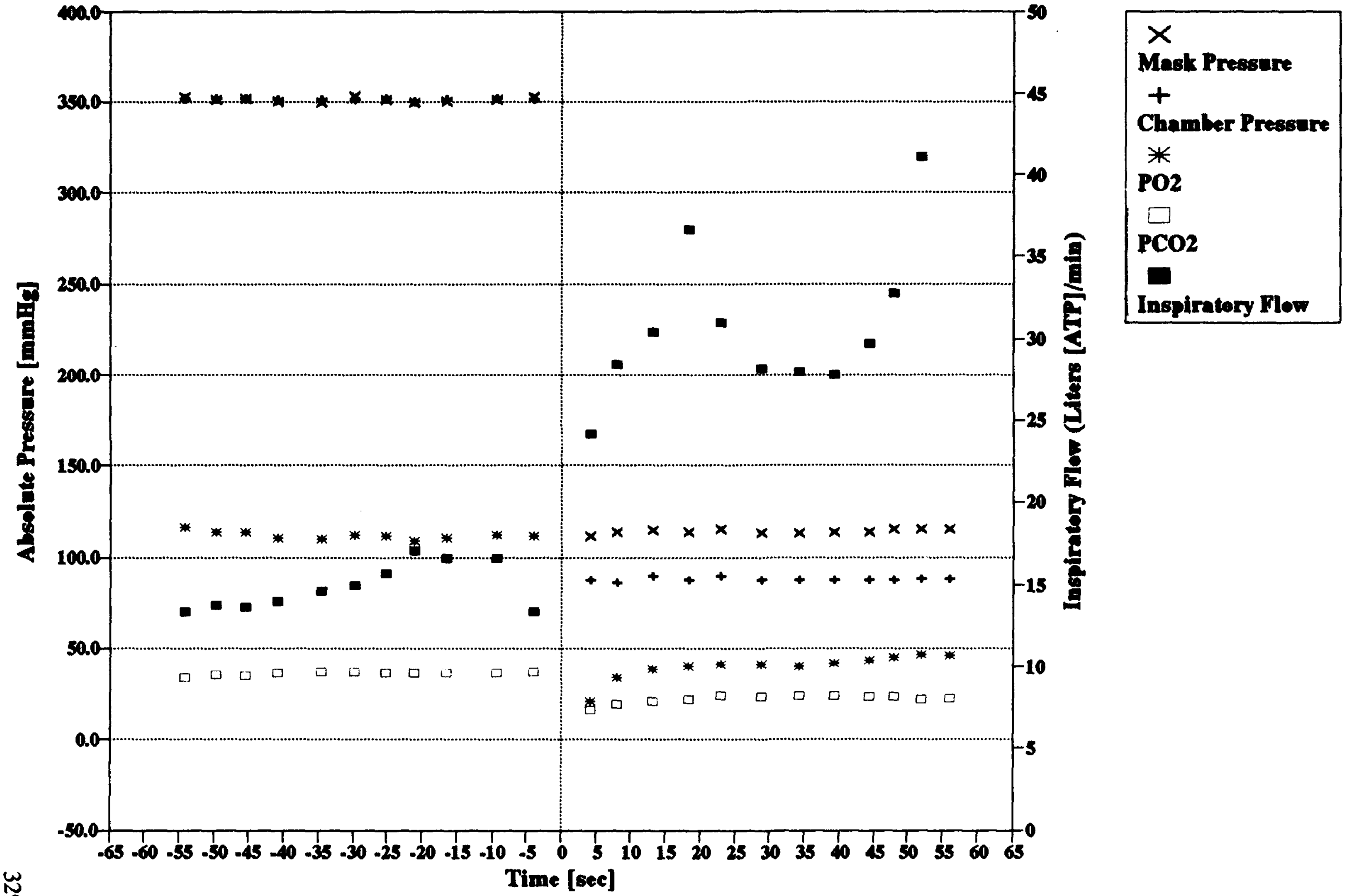
Subject: SHI / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



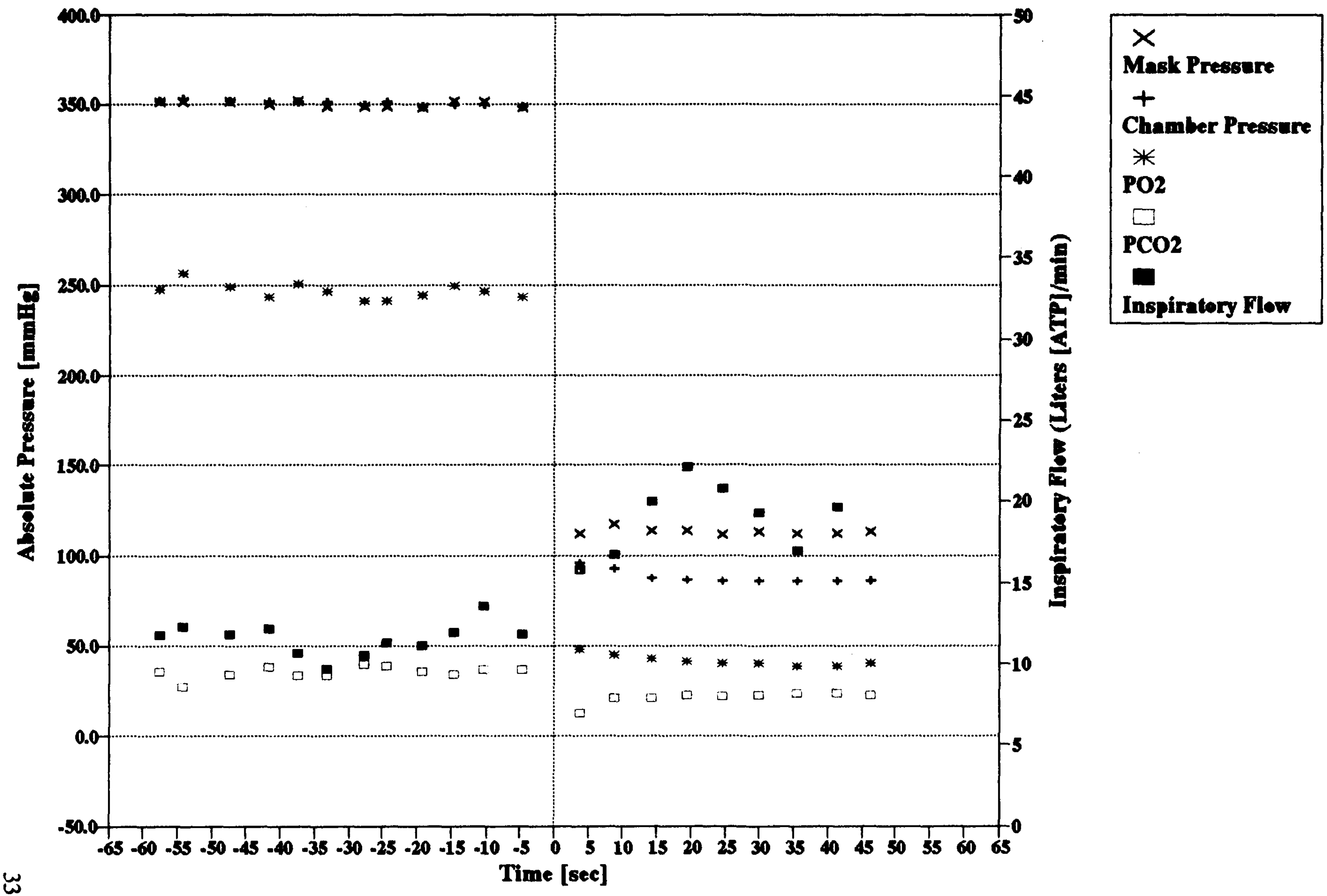
Subject: SHN / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



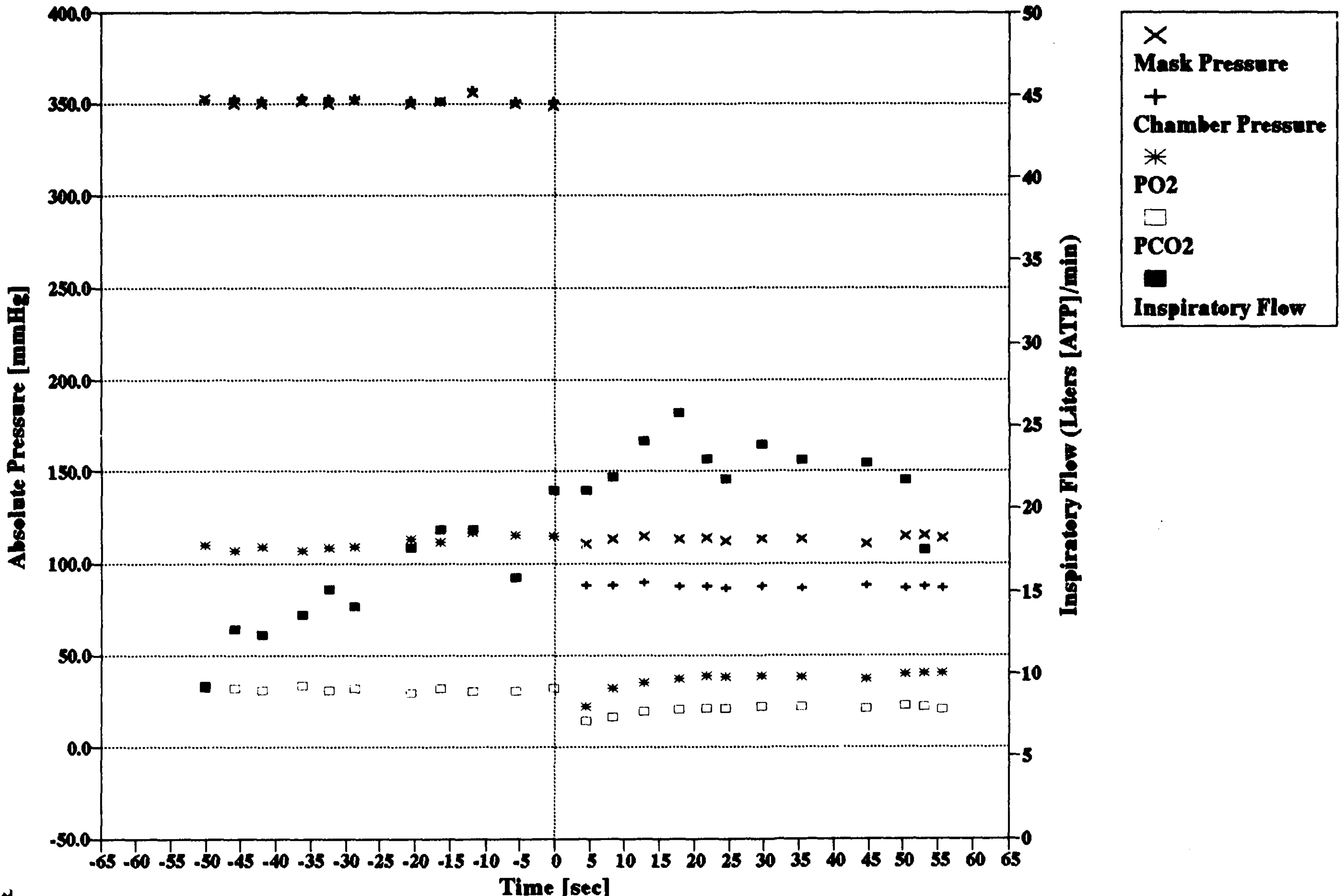
Subject: SHN / 100% O2 Dilution
20/50 kft Rapid Decompression



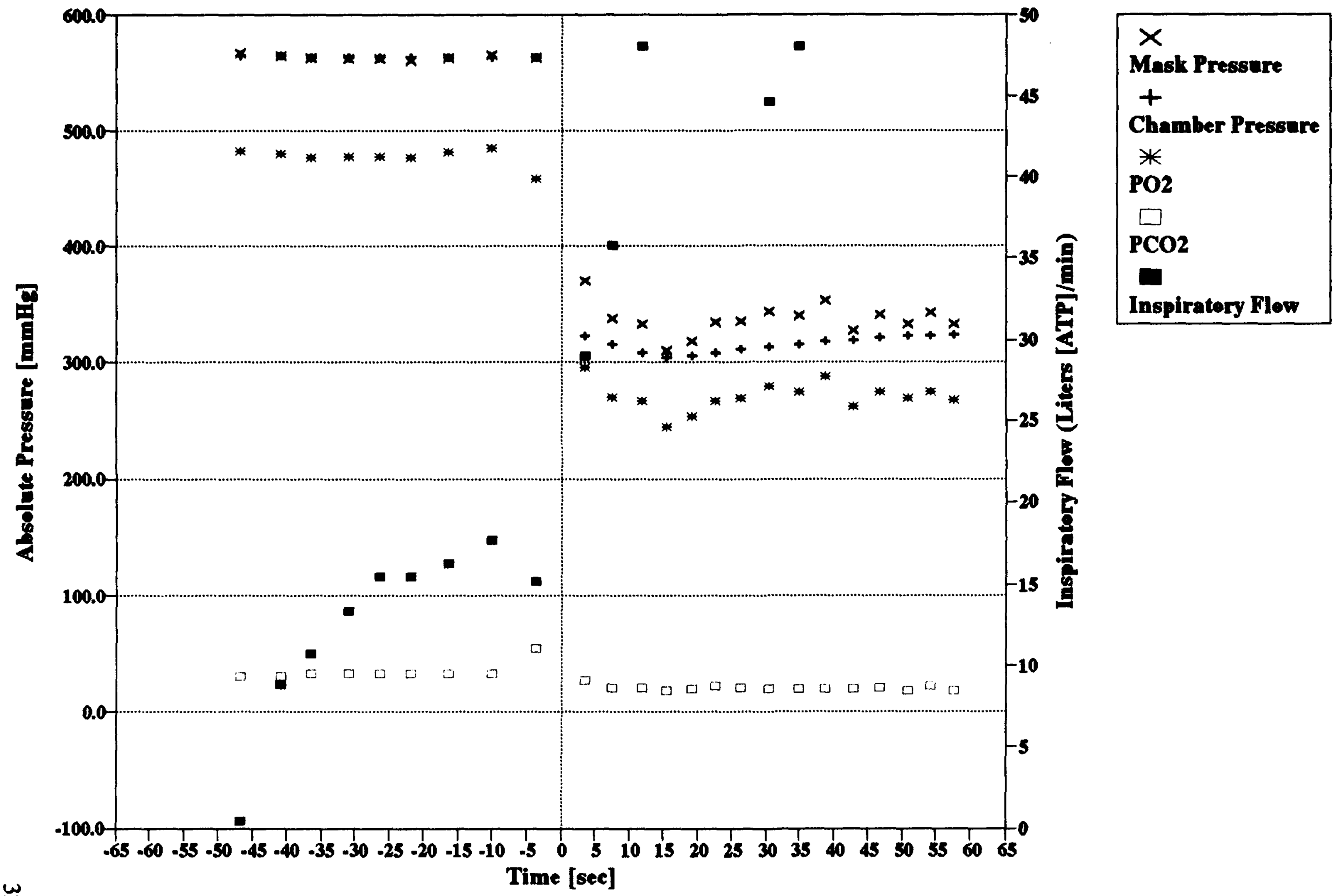
Subject: SHN / 93% O2 Non-Dilution
20/50 kft Rapid Decompression



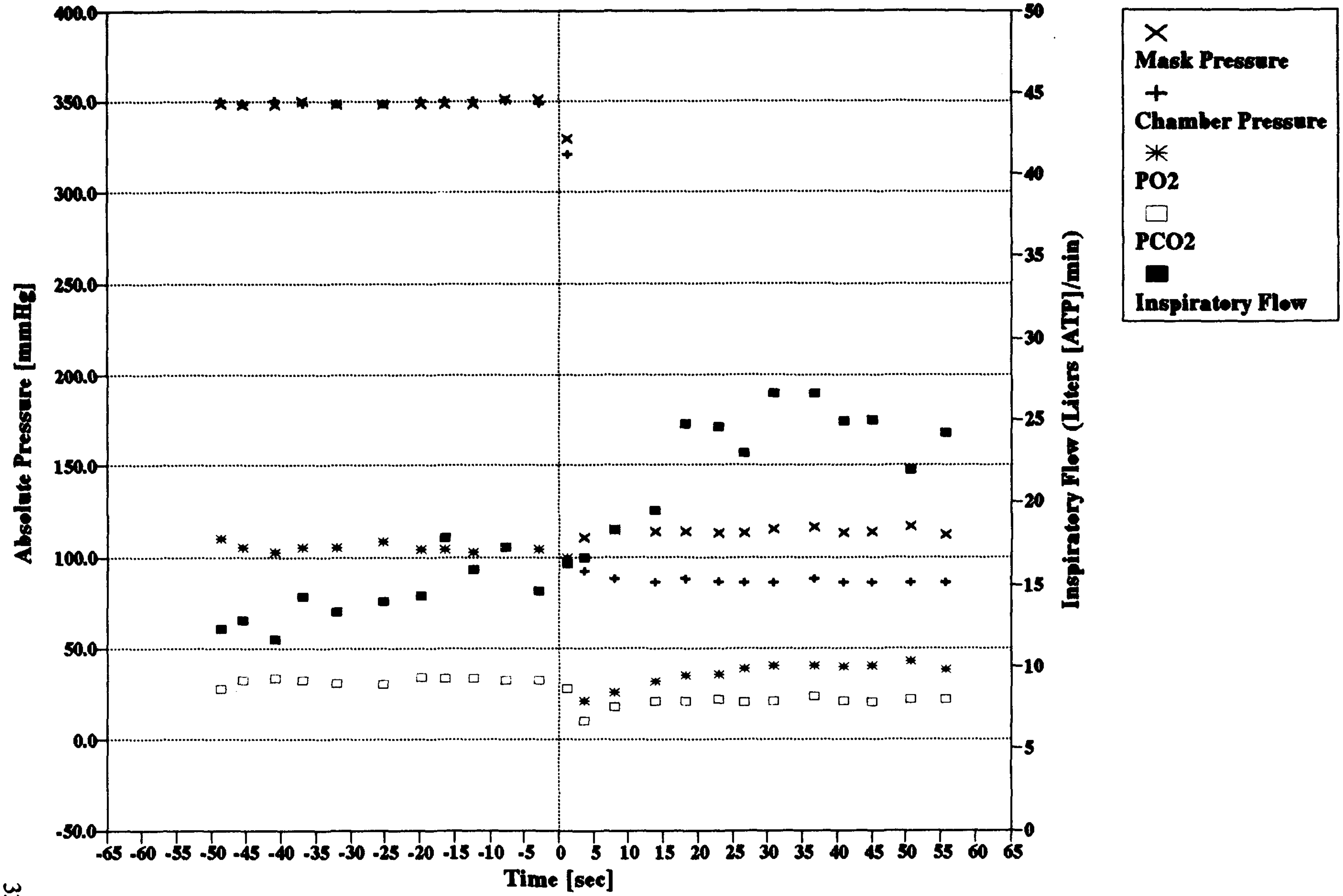
Subject: SHN / 93% O2 Dilution
20/50 kft Rapid Decompression



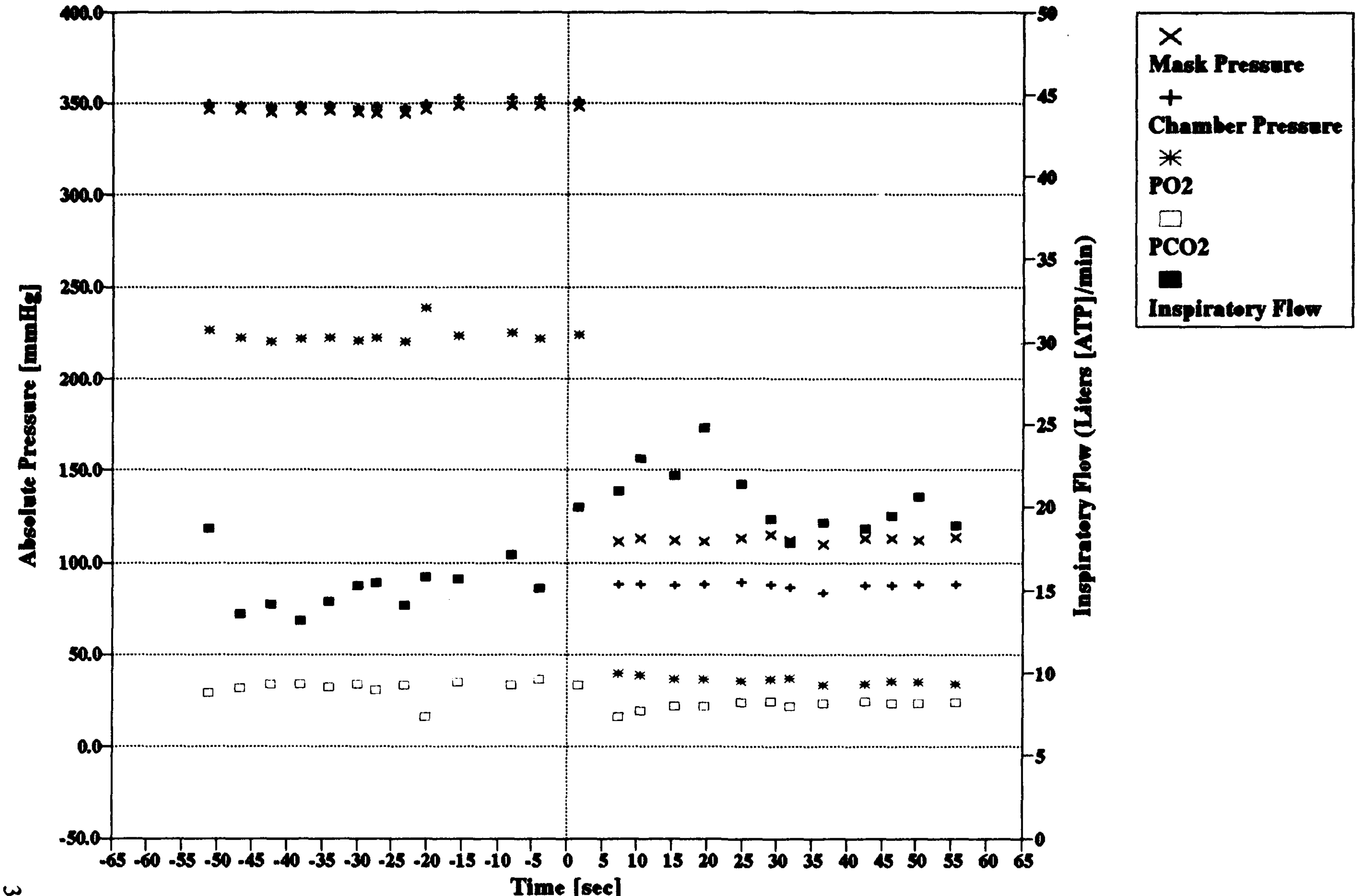
Subject: SHN / 90% O2 Non-Dilution
8/20 kft Rapid Decompression



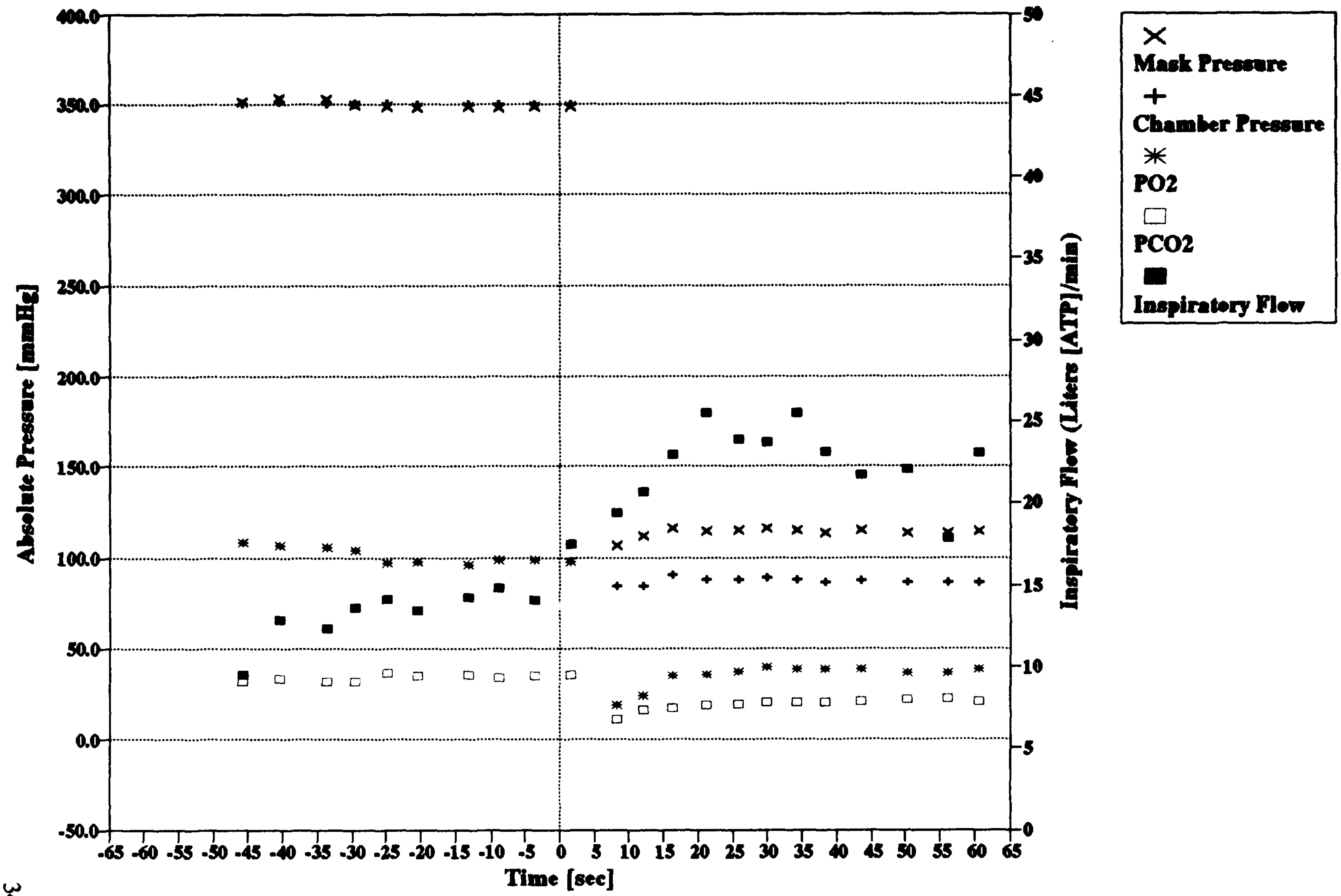
Subject: SHN / 90% O2 Dilution
20/50 kft Rapid Decompression



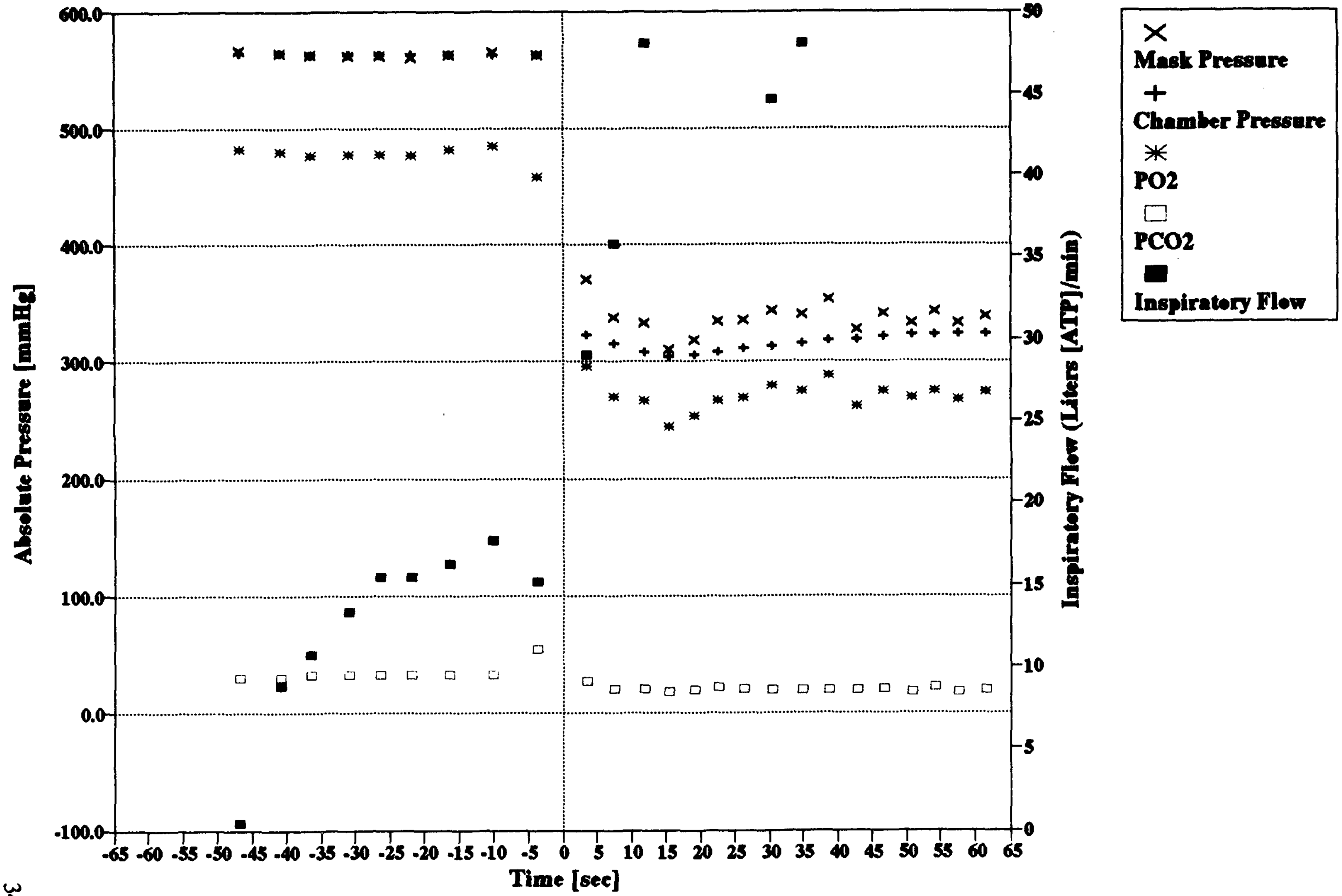
Subject: SHN / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



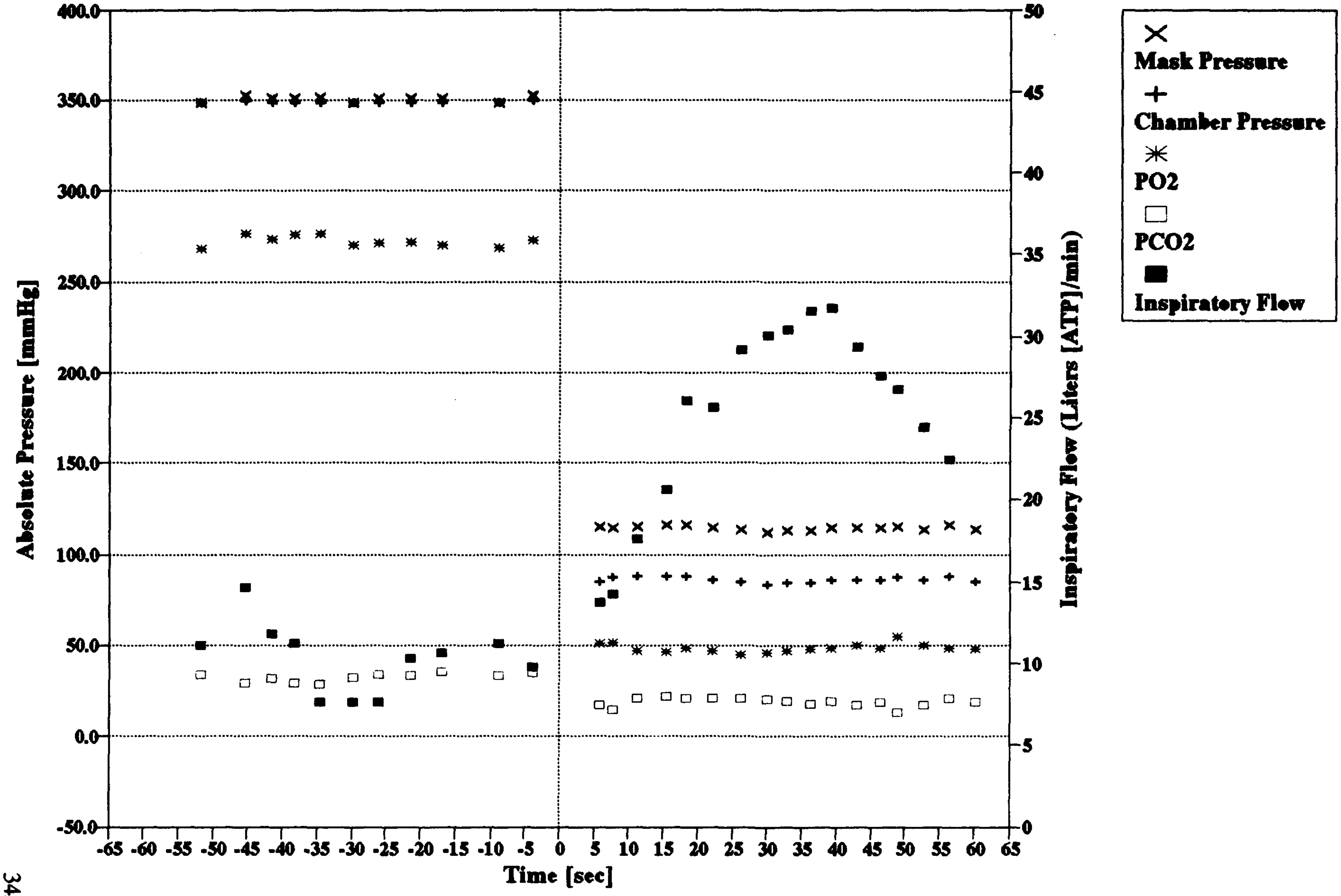
Subject: SHN / 85% O2 Dilution
20/50 kft Rapid Decompression



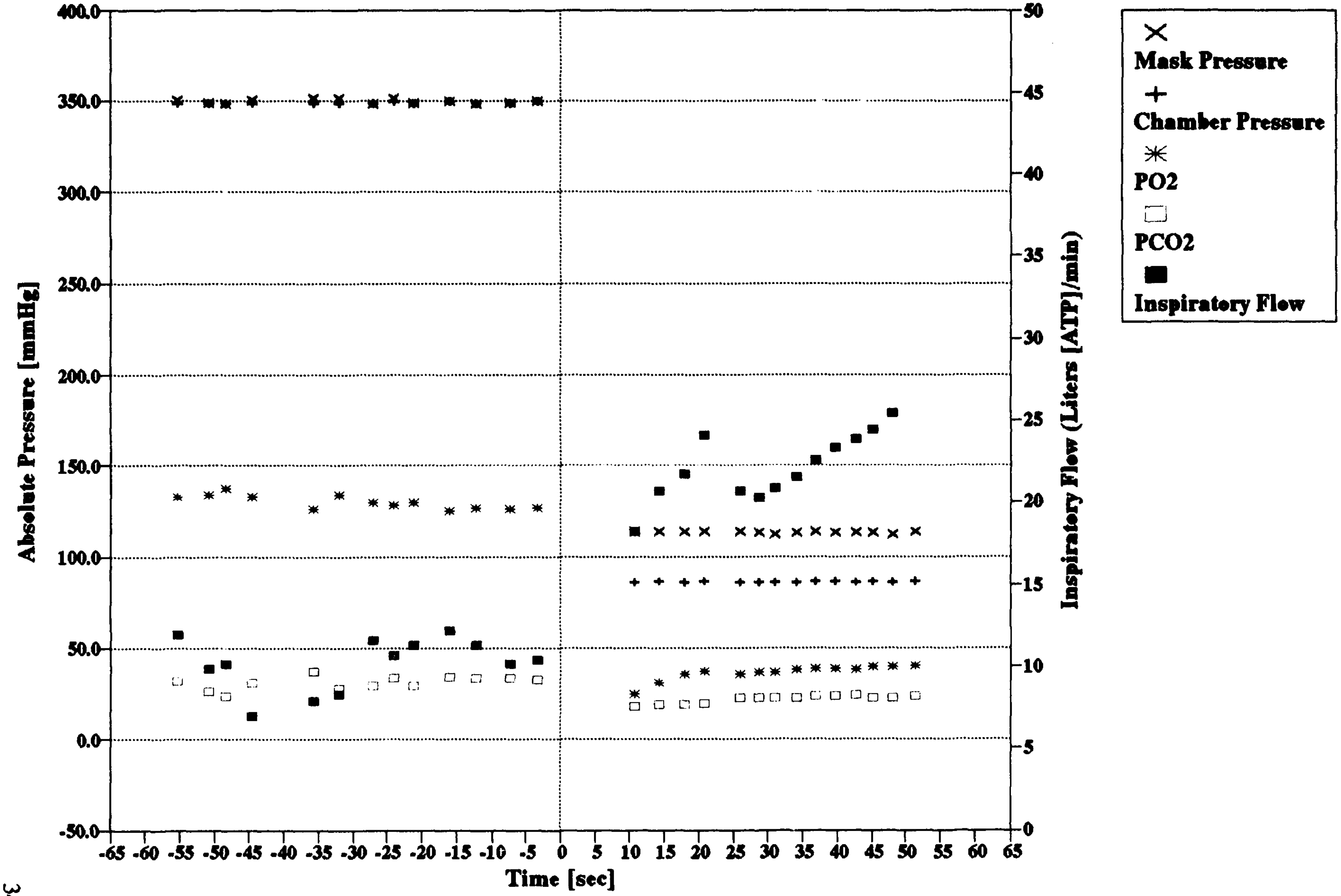
Subject: SHN / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



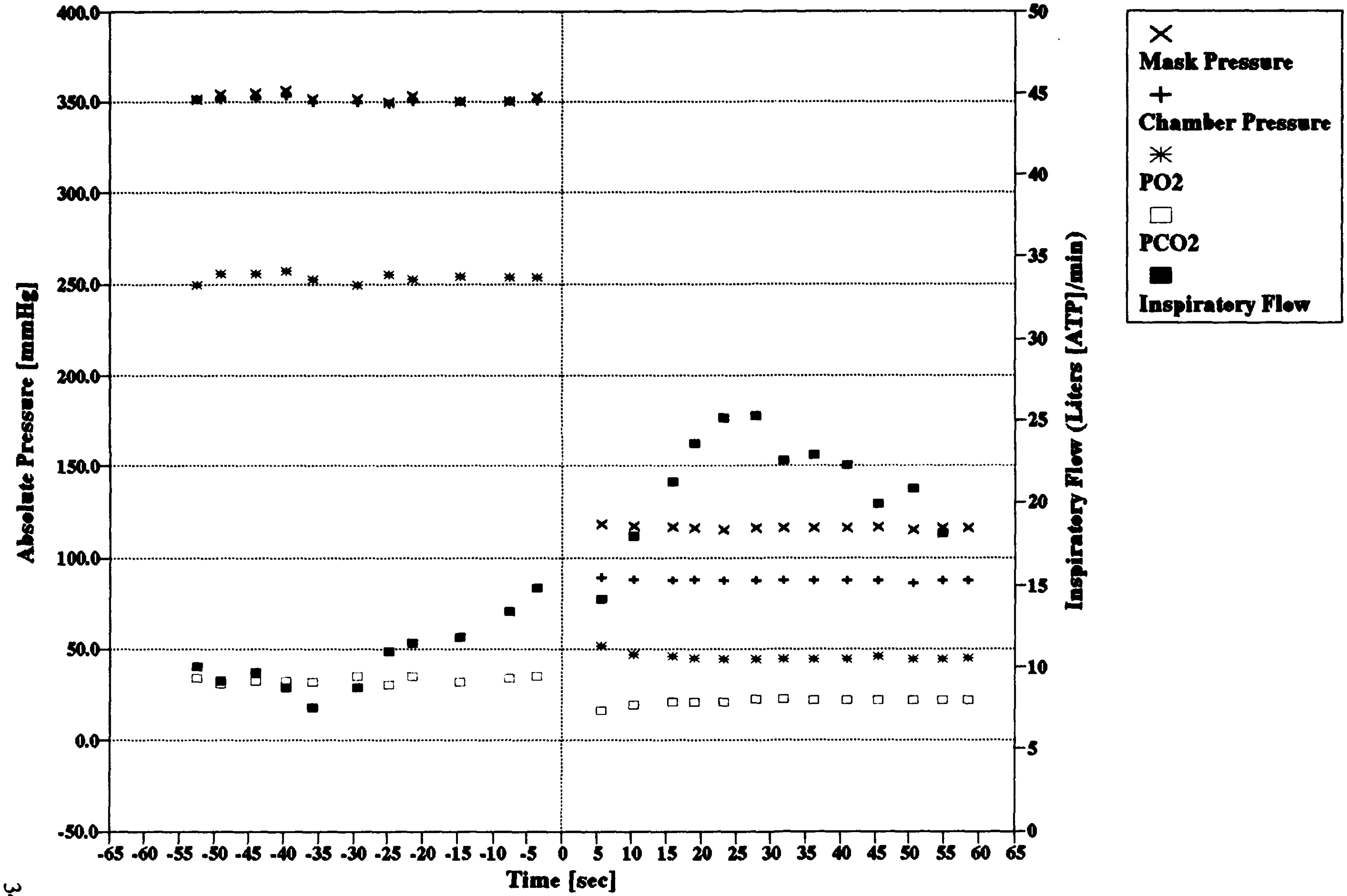
Subject: TED / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



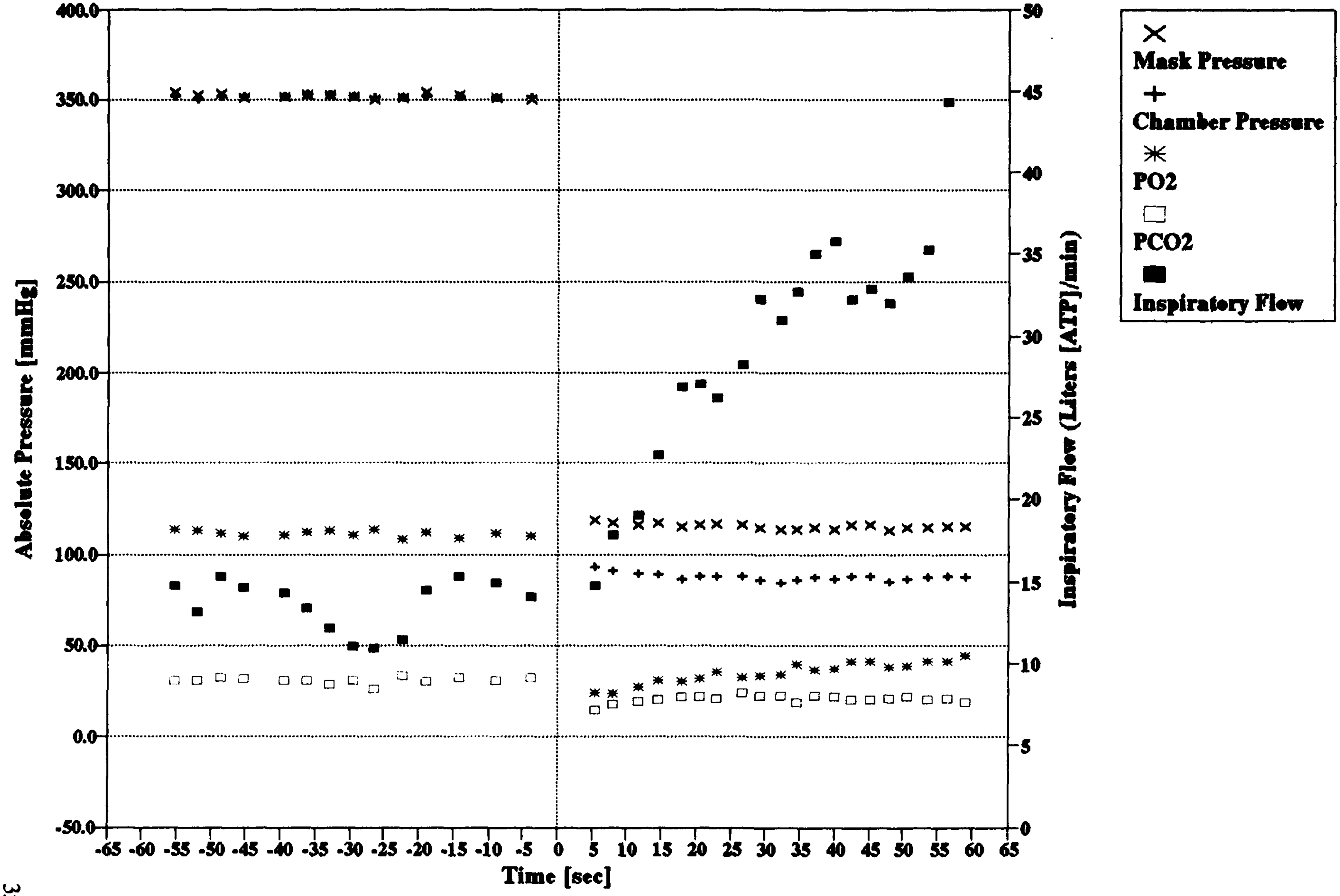
Subject: TED / 100% O2 Dilution
20/50 kft Rapid Decompression



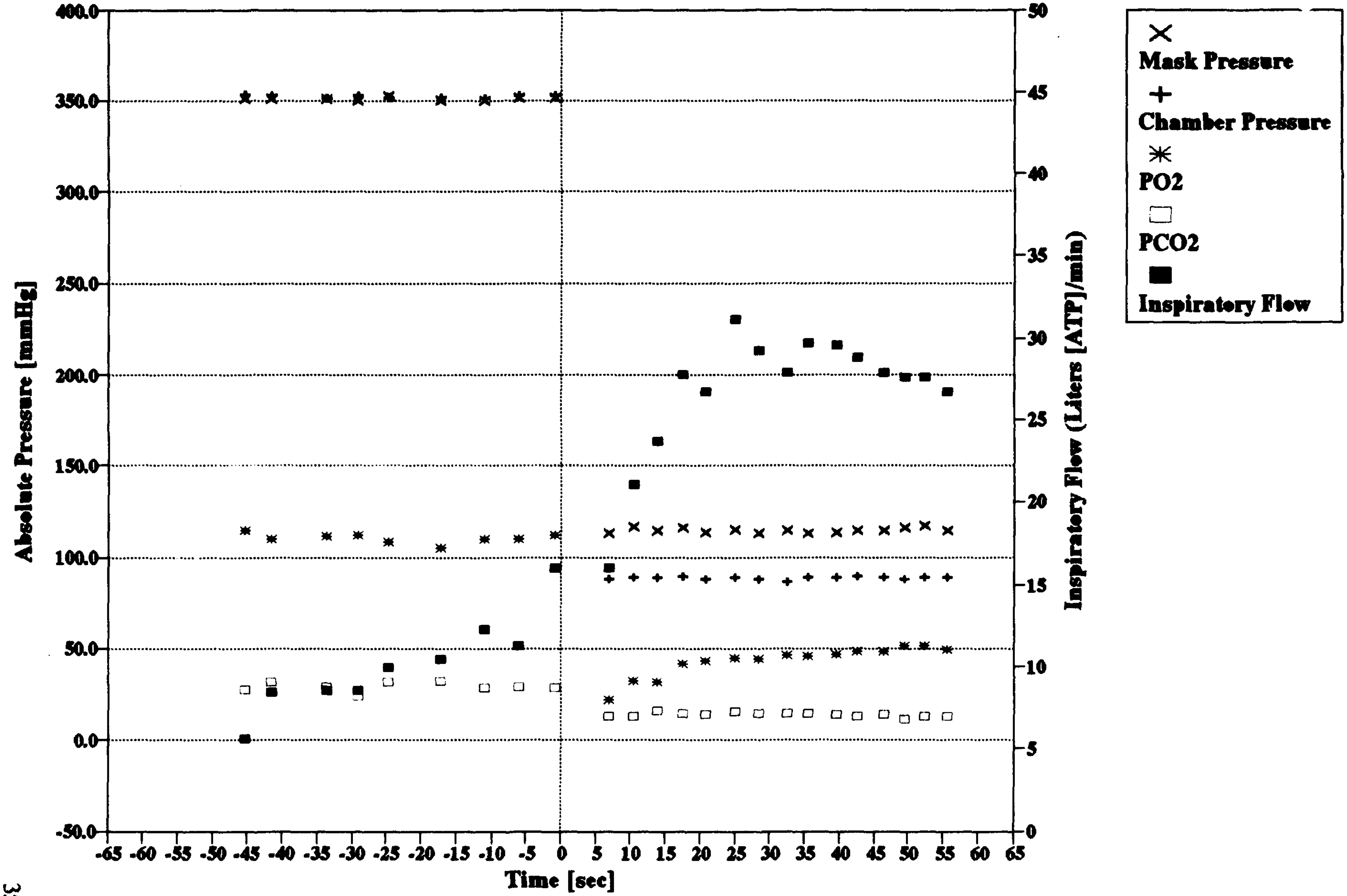
Subject: TED / 93% O2 Non-Dilution
20/50 kft Rapid Decompression



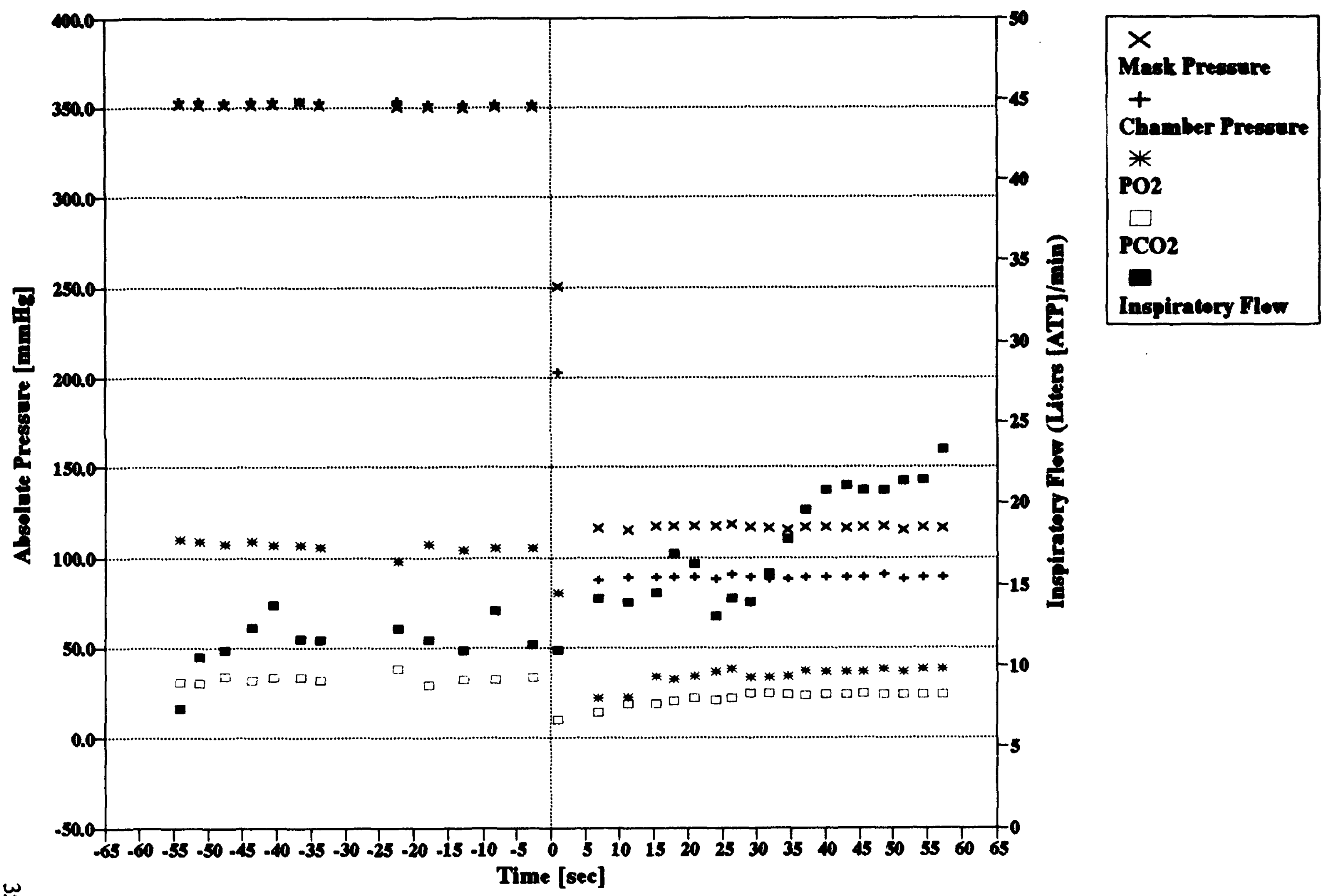
Subject: TED / 93% O2 Dilution
20/50 kft Rapid Decompression

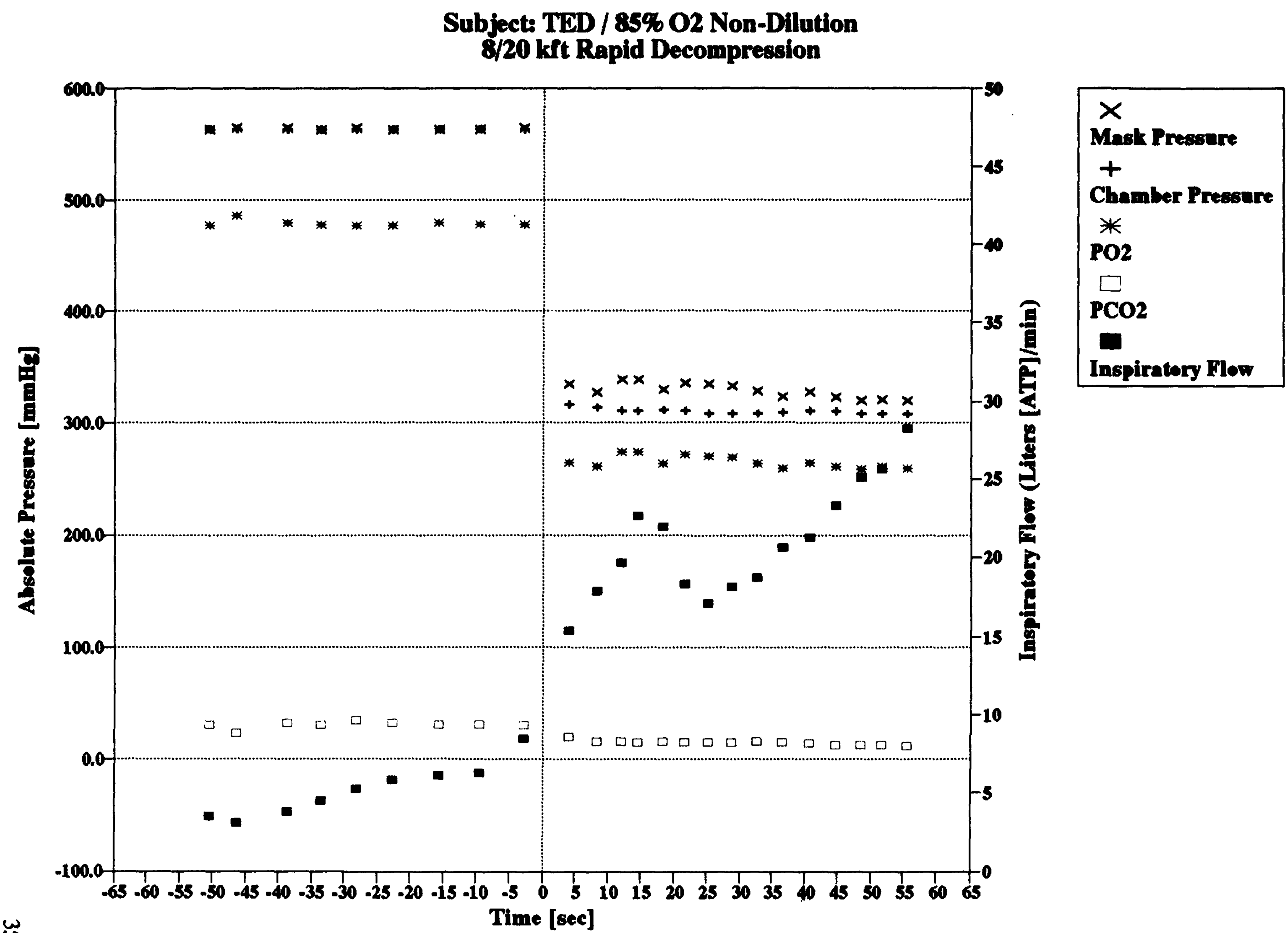


Subject: TED / 90% O2 Non-Dilution
20/50 kft Rapid Decompression

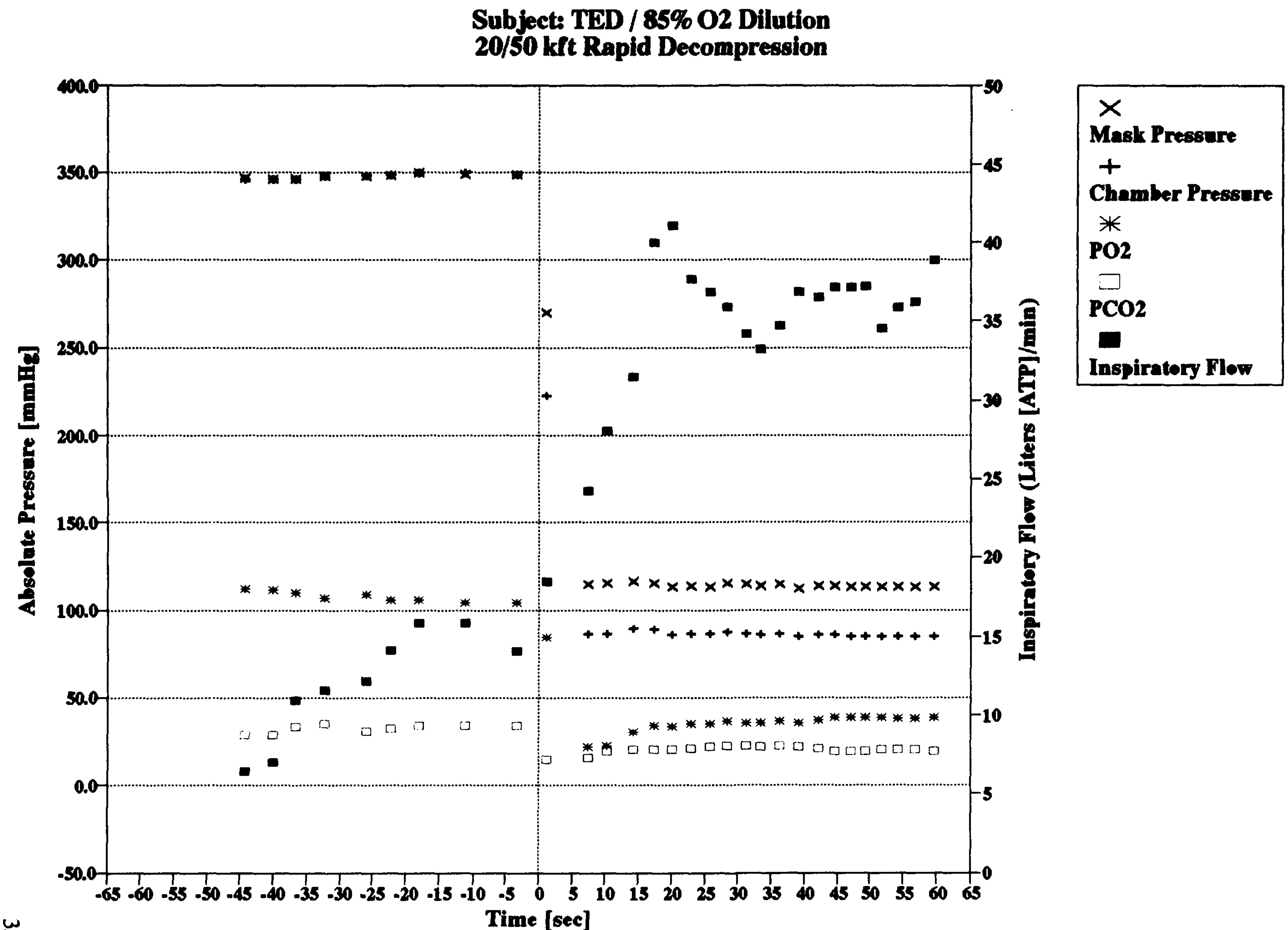


Subject: TED / 90% O2 Dilution
20/50 kft Rapid Decompression



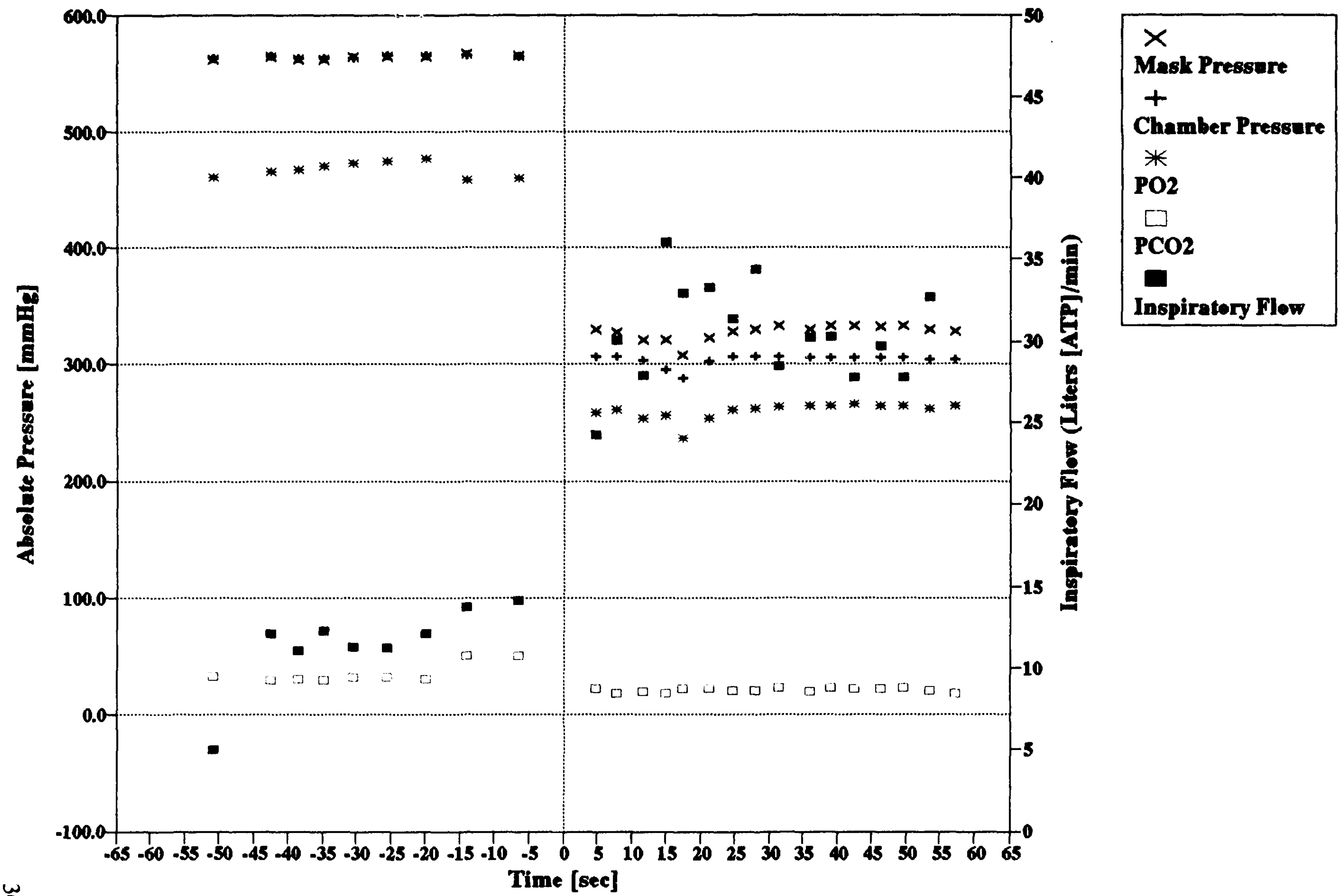


Subject: TED / 85% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

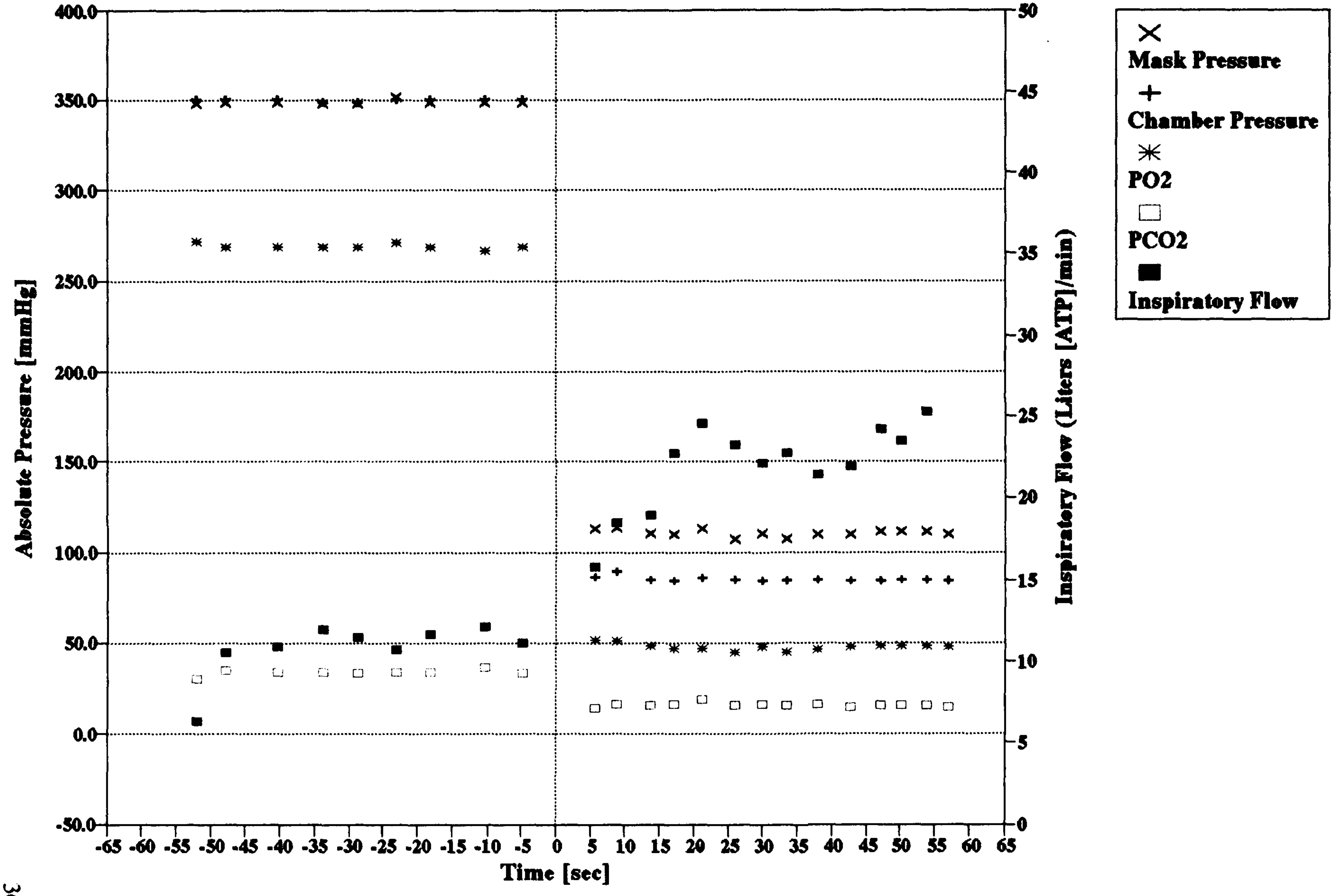


Subject: TED / 85% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

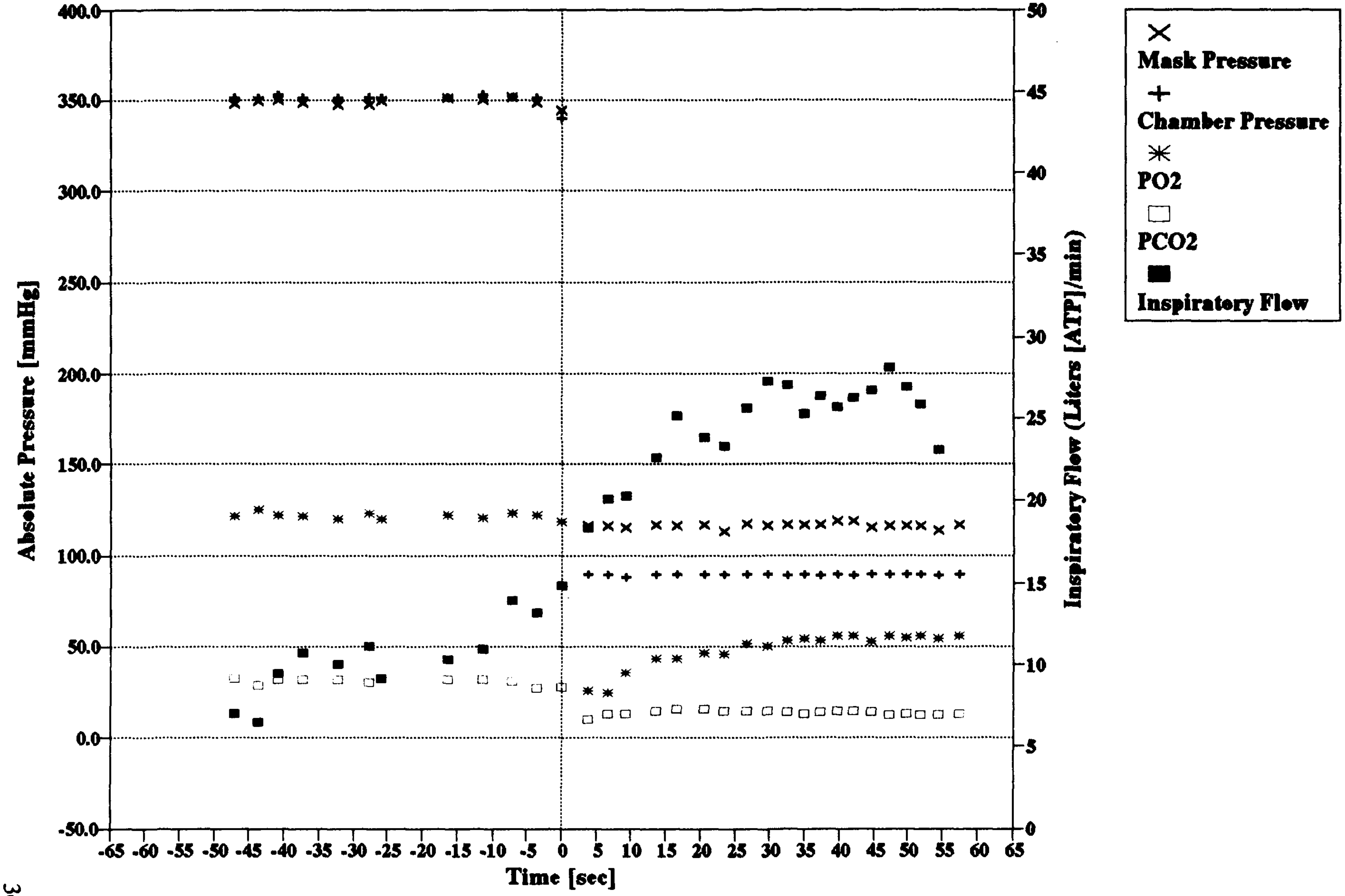
Subject: TED / 100% O2 Non-Dilution
8/20 kft Rapid Decompression



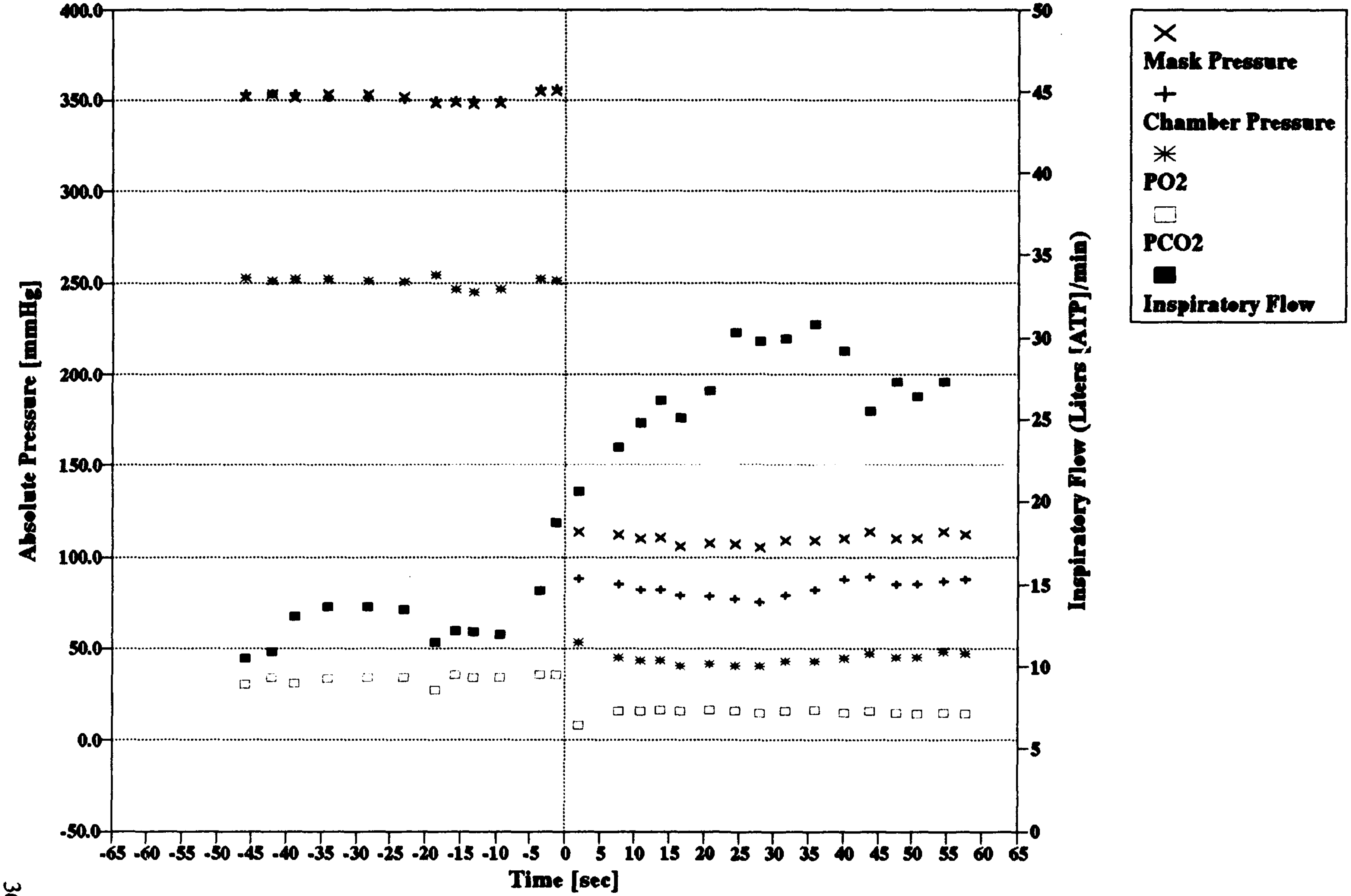
Subject: WRI / 100% O2 Non-Dilution
20/50 kft Rapid Decompression



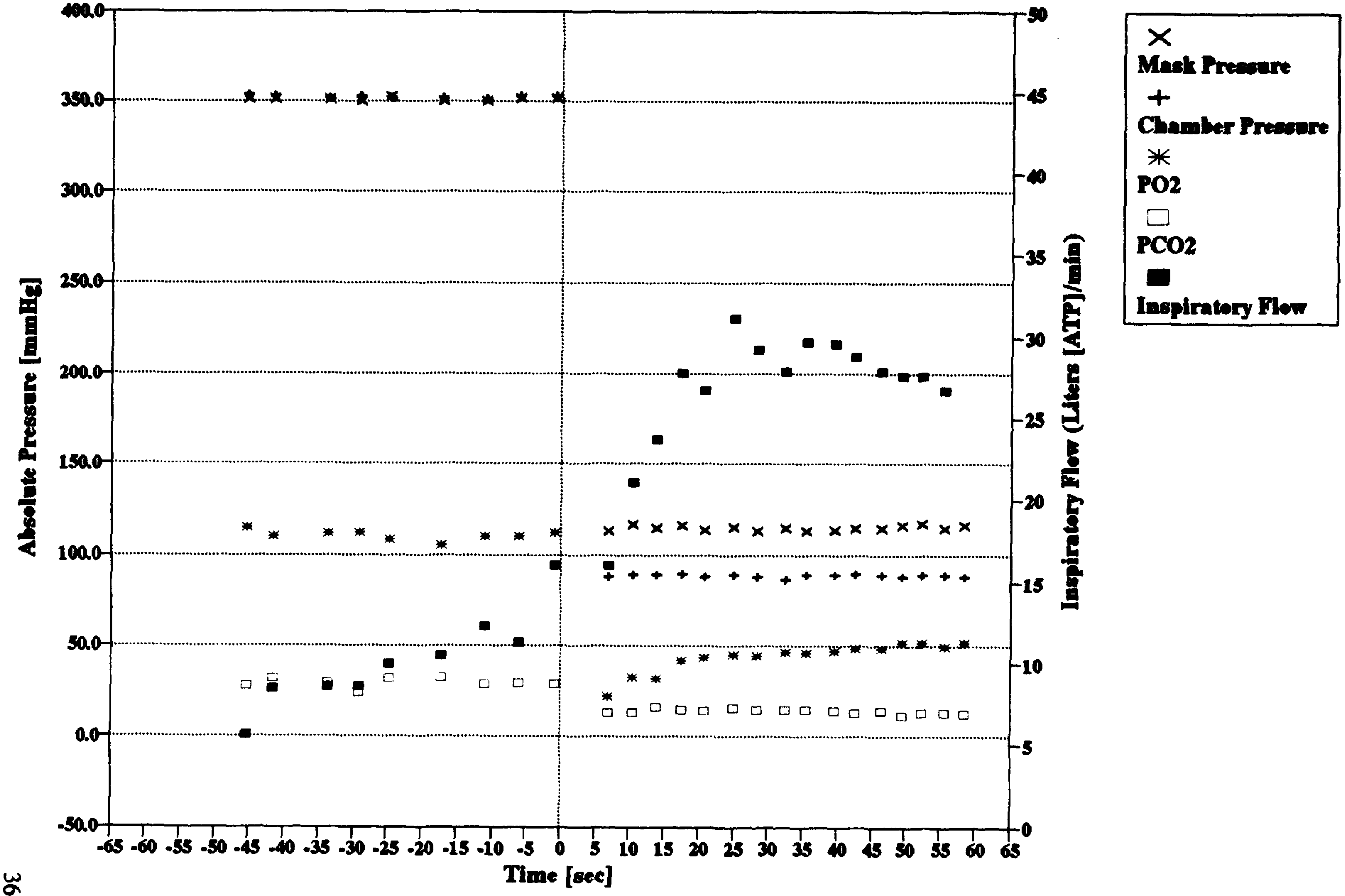
Subject: WRI / 100% O2 Dilution
20/50 kft Rapid Decompression



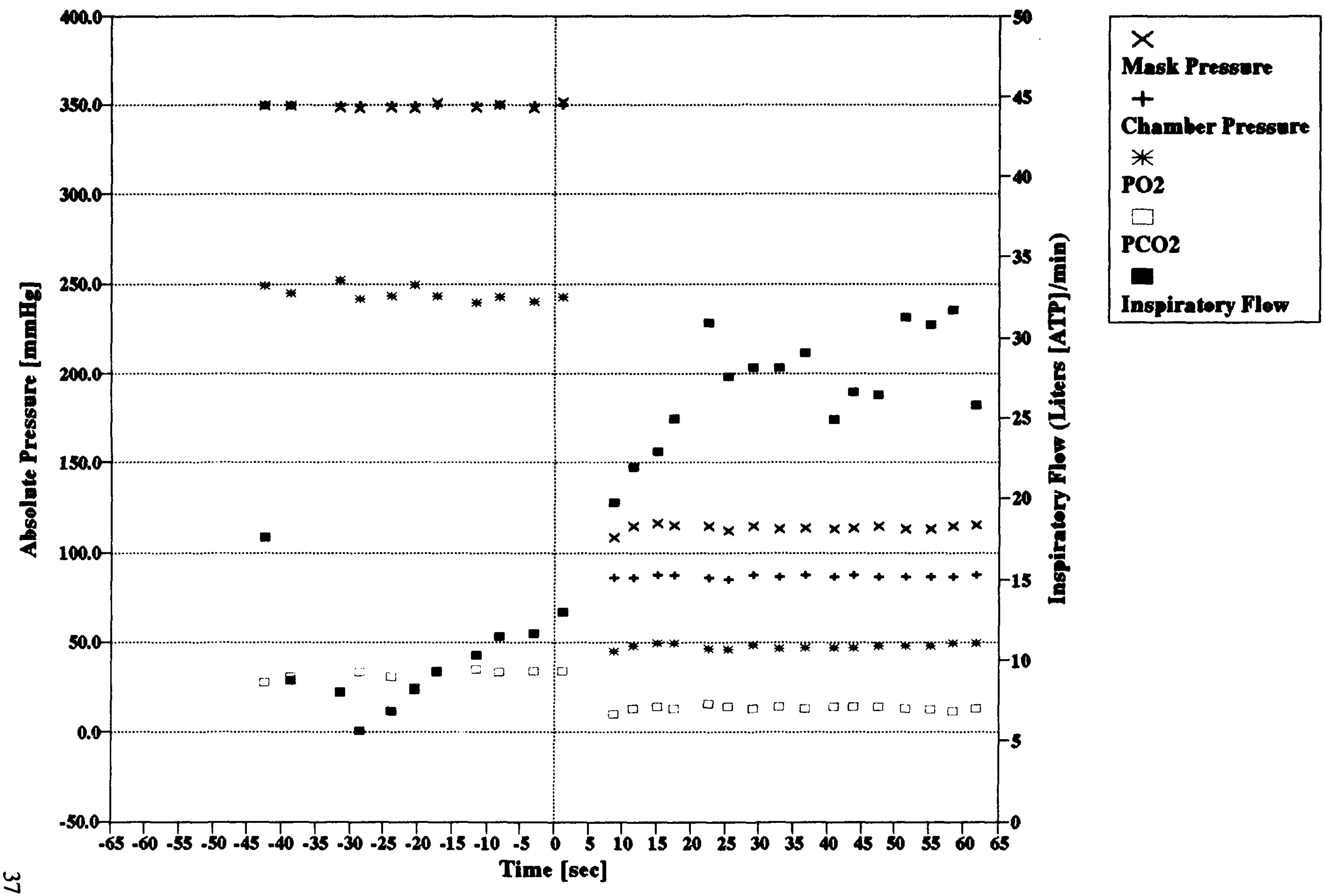
Subject: WRI / 93% O2 Non-Dilution
20/50 kft Rapid Decompression



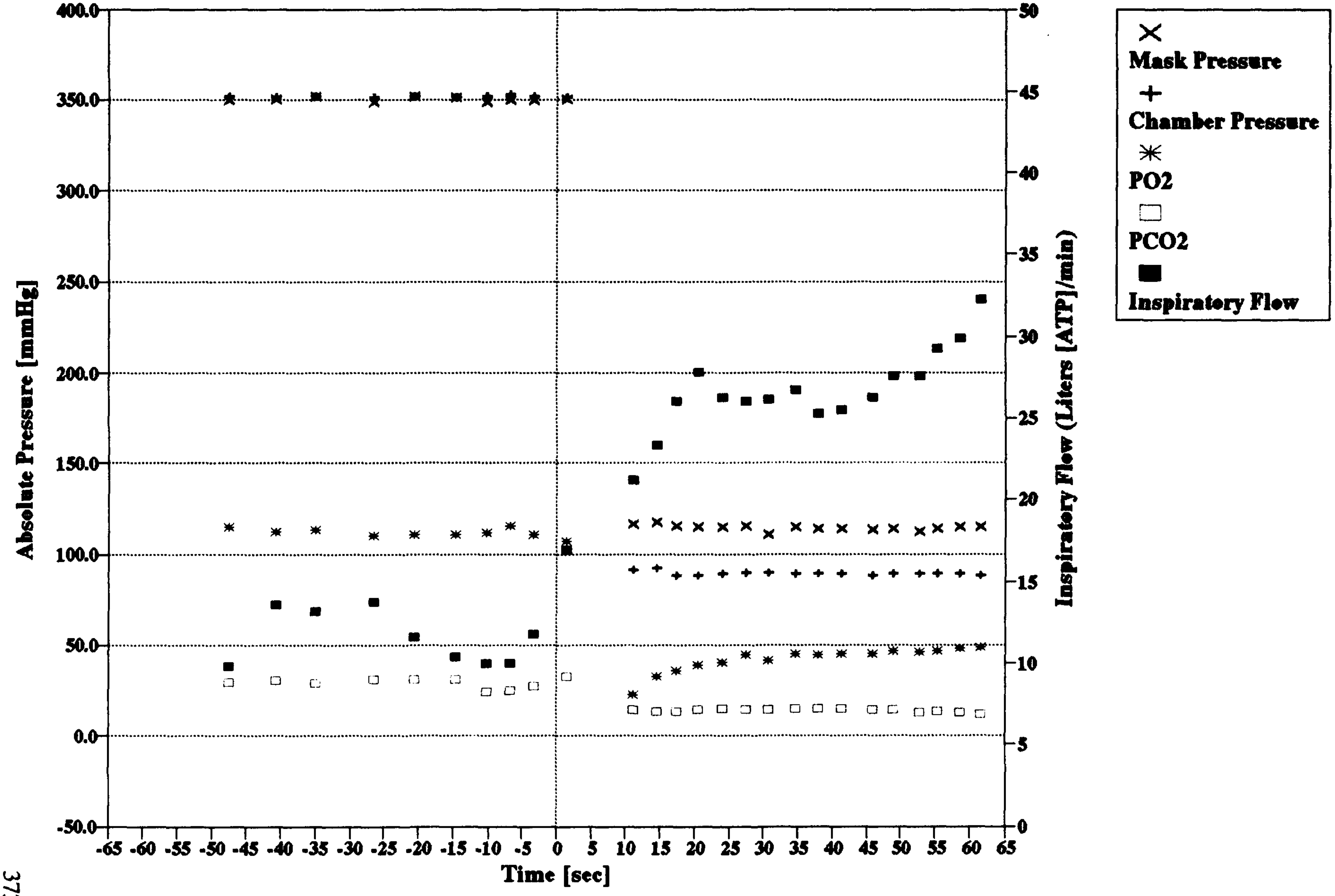
Subject: WRI / 93% O2 Dilution
20/50 kft Rapid Decompression



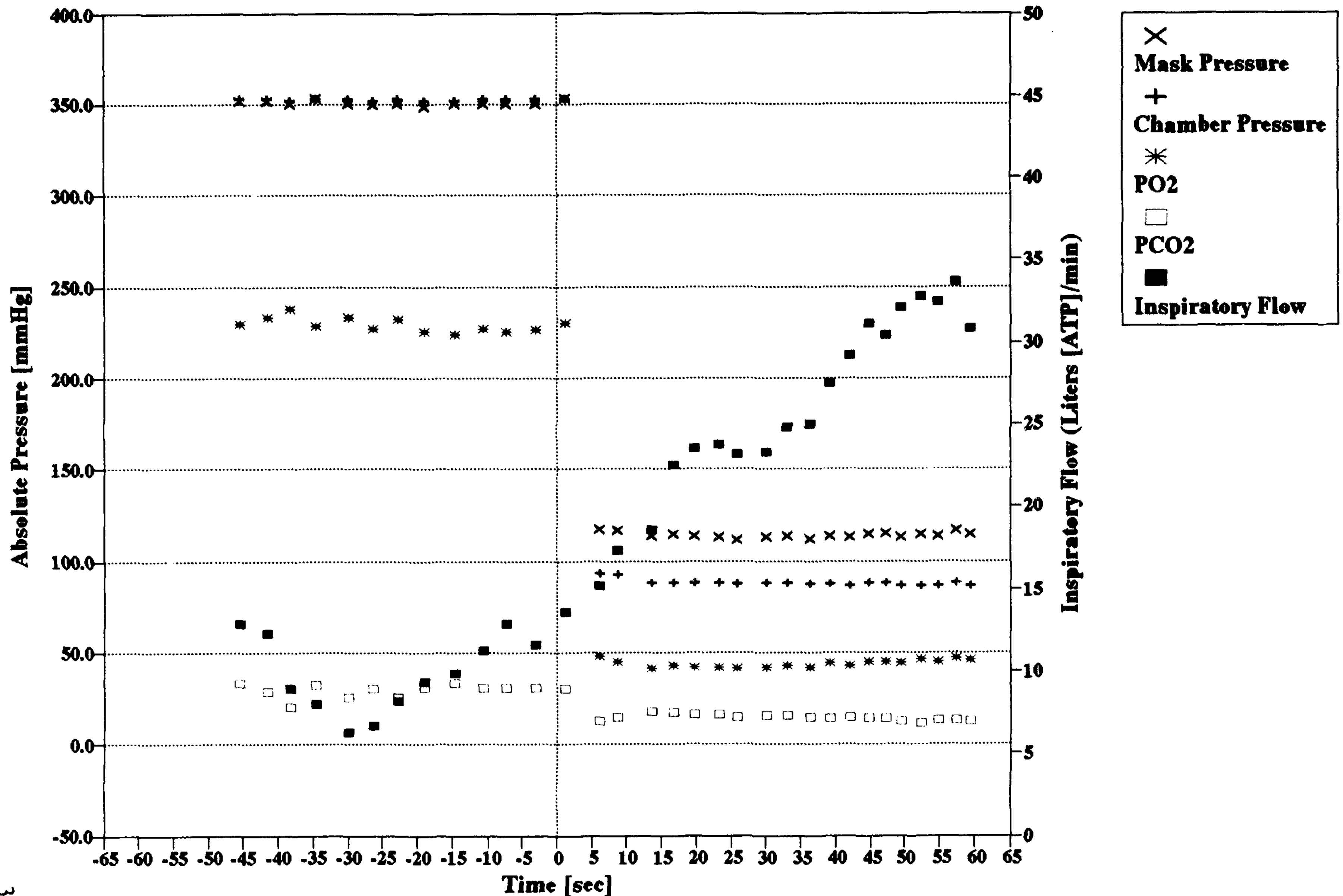
Subject: WRI / 90% O2 Non-Dilution
20/50 kft Rapid Decompression

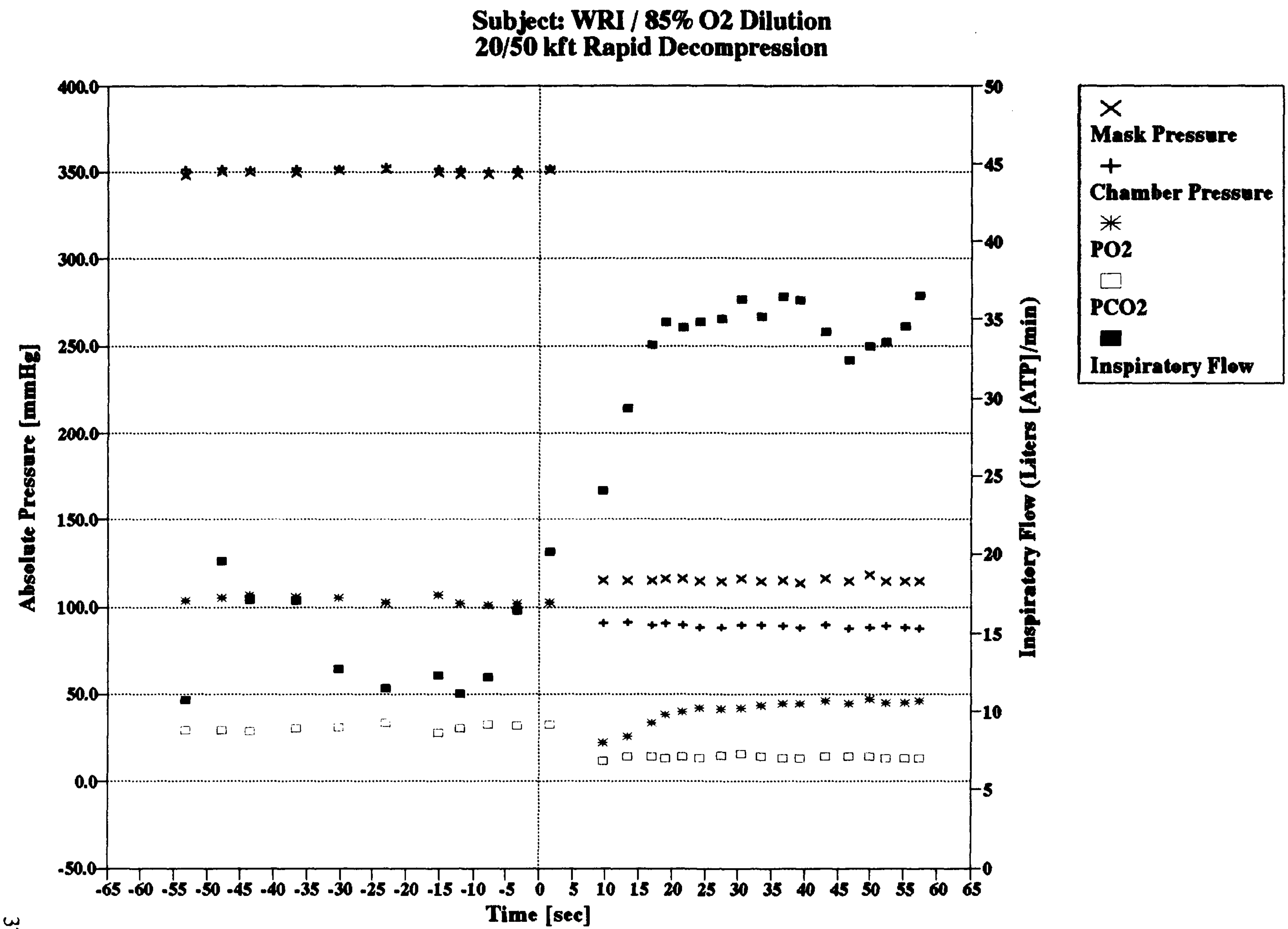


Subject: WRI / 90% O2 Dilution
20/50 kft Rapid Decompression

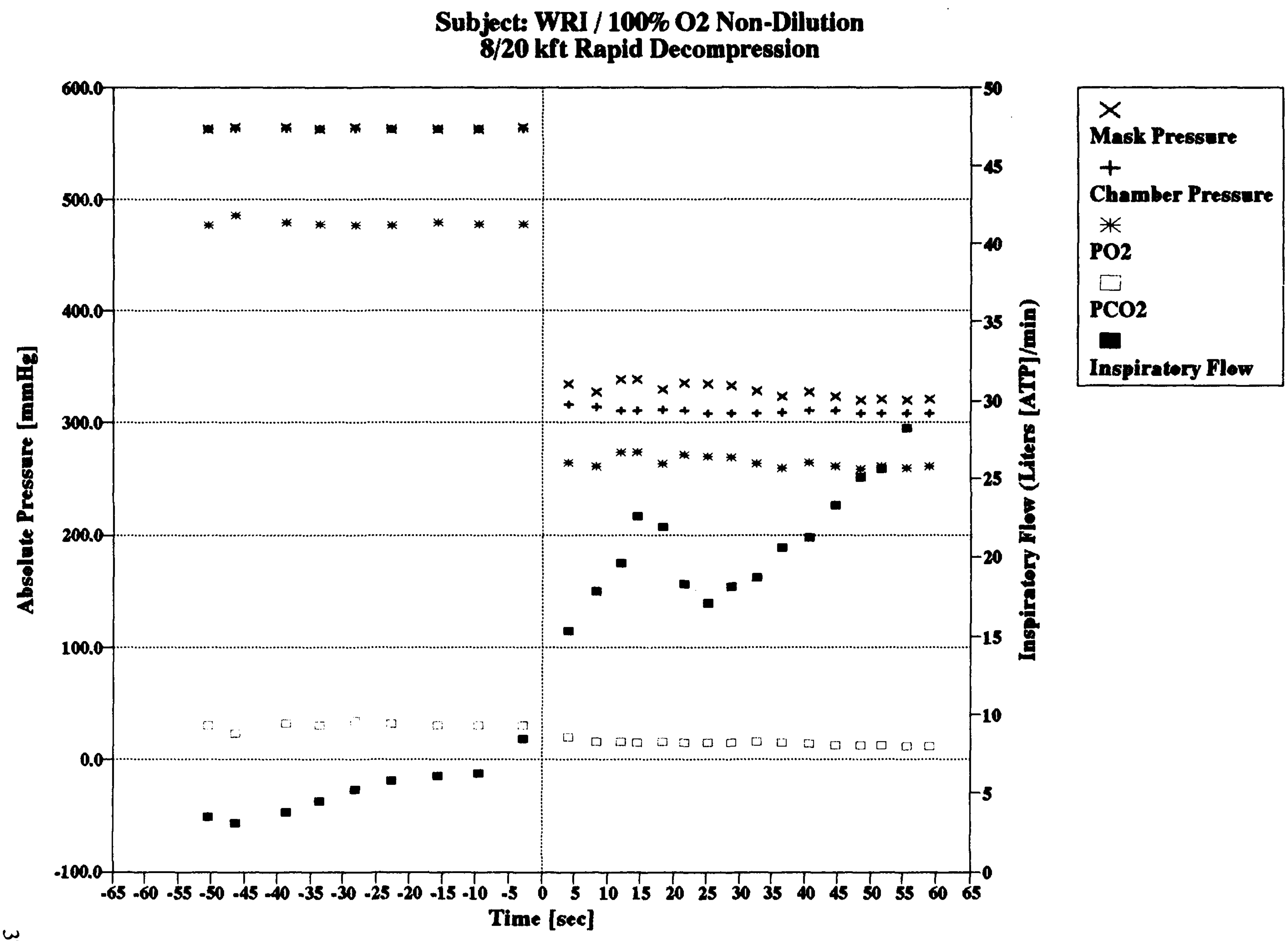


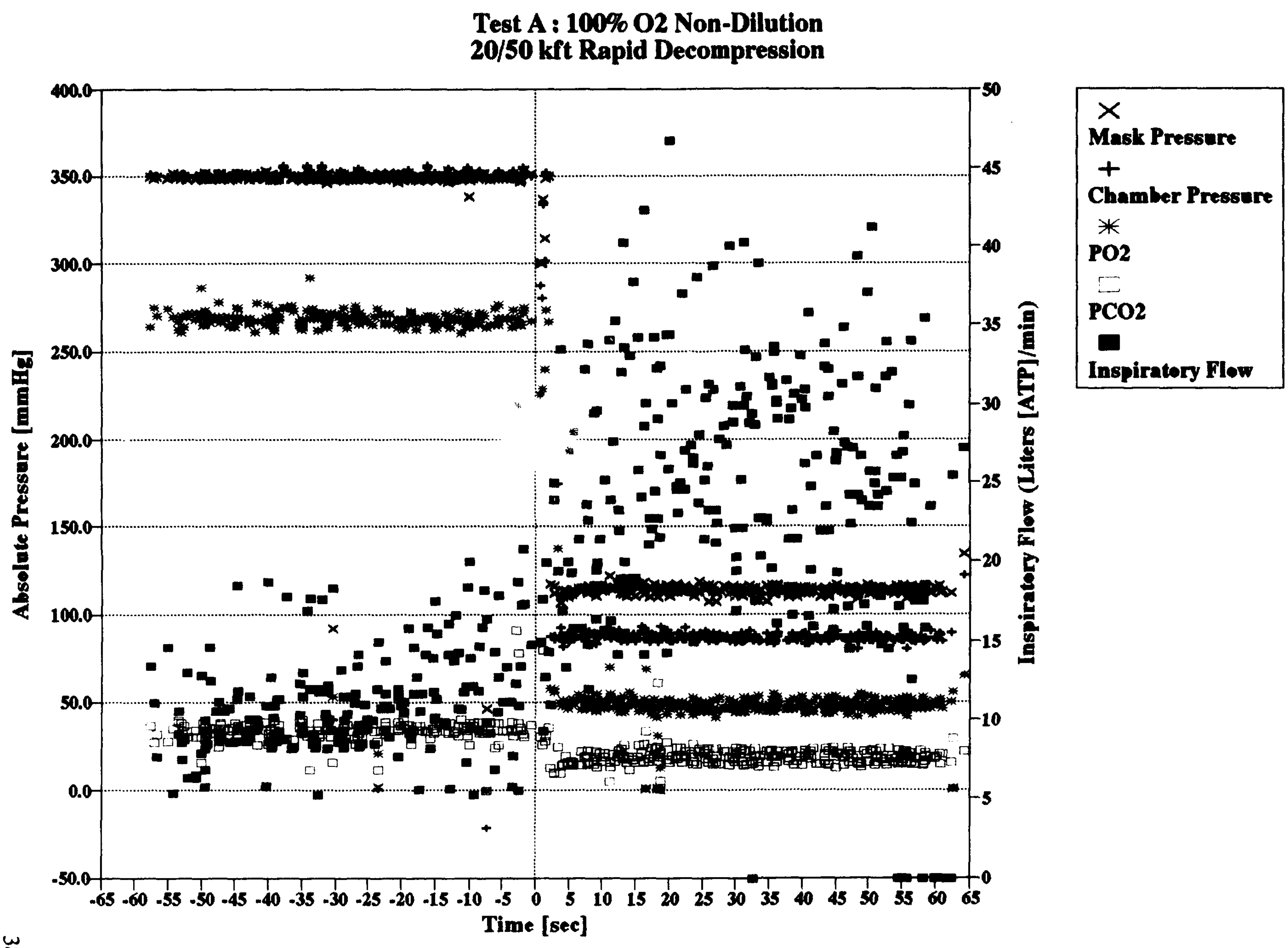
Subject: WRI / 85% O2 Non-Dilution
20/50 kft Rapid Decompression



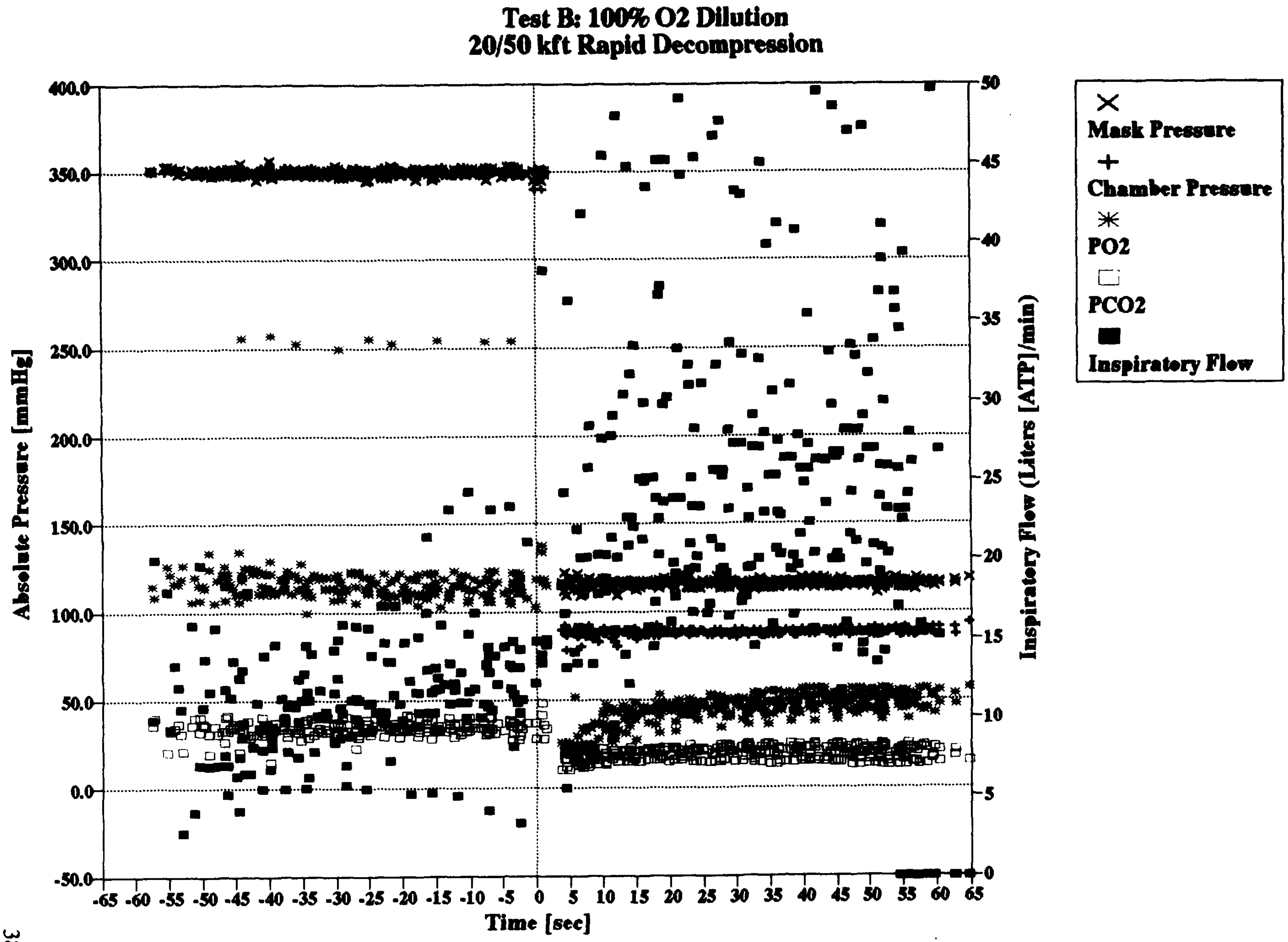


Subject: WRI / 85% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
Inspiratory Flow (Liters [ATP]/min)
Time [sec]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow


Subject: WRI / 100% O2 Non-Dilution
8/20 kft Rapid Decompression
Absolute Pressure [mmHg]
600.0
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

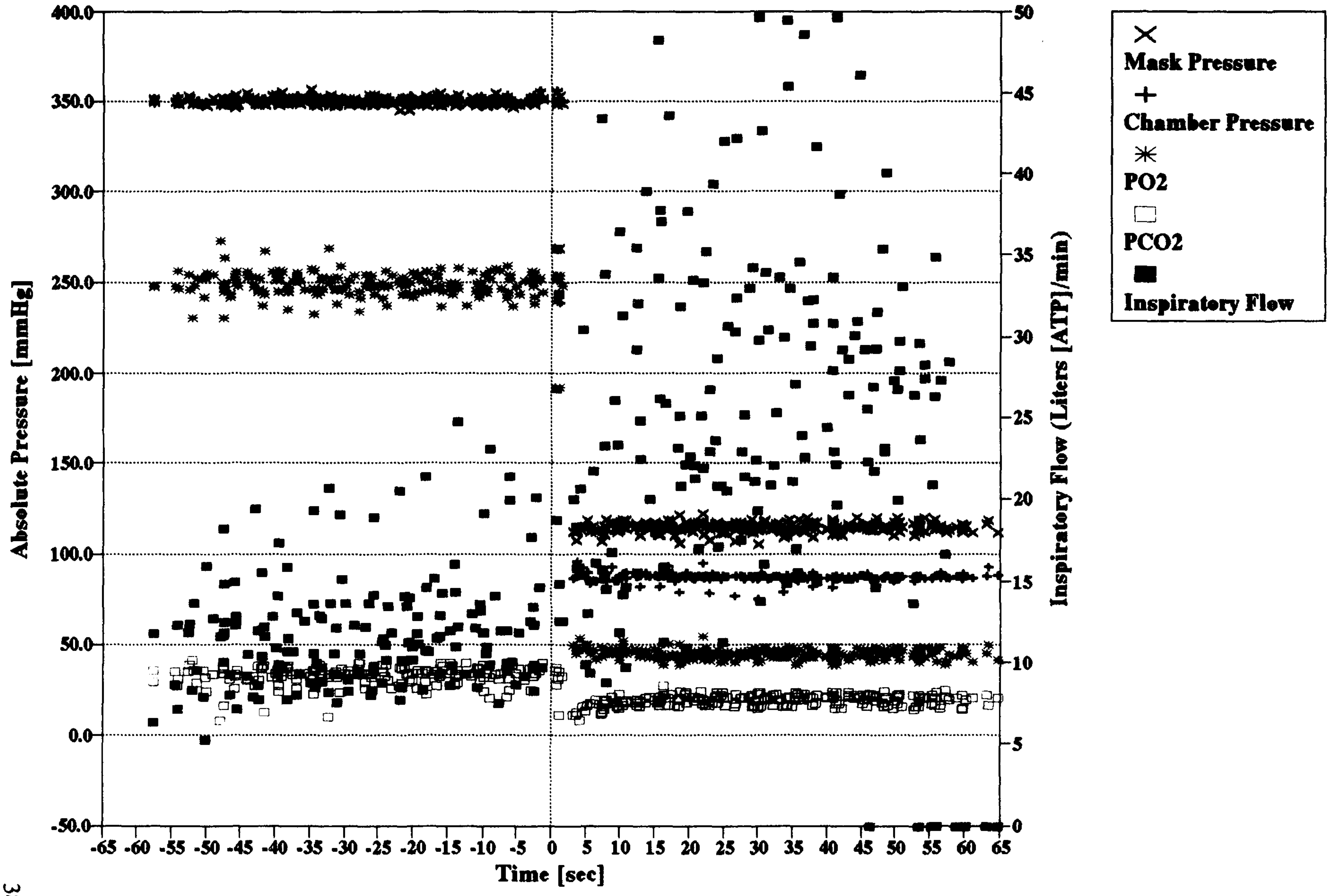


Test A : 100% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Inspiratory Flow (Liters [ATP]/min)
50 45 40 35 30 25 20 15 10 5 0
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

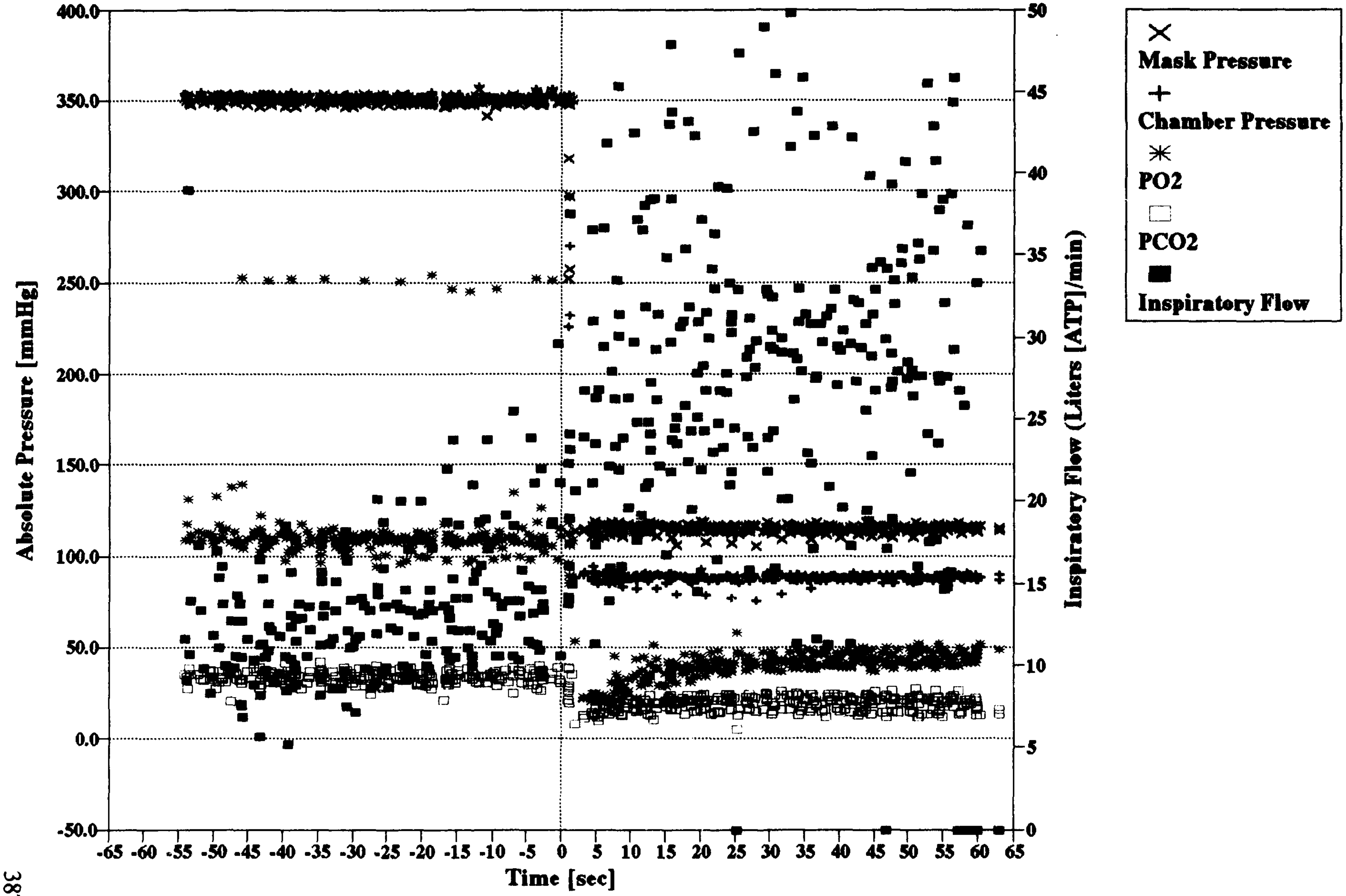


Test B: 100% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

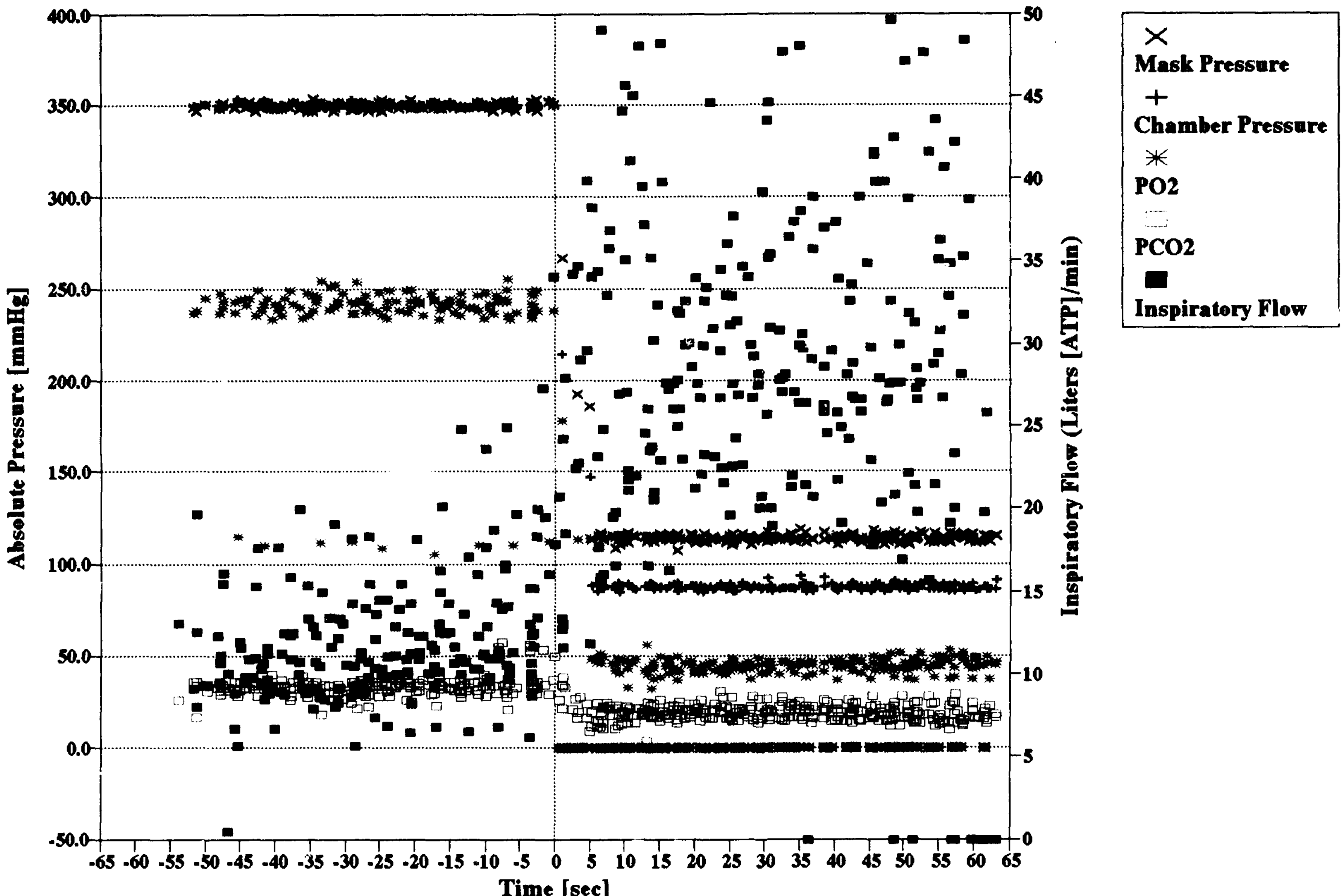
Test C: 93% O2 Non-Dilution
20/50 kft Rapid Decompression

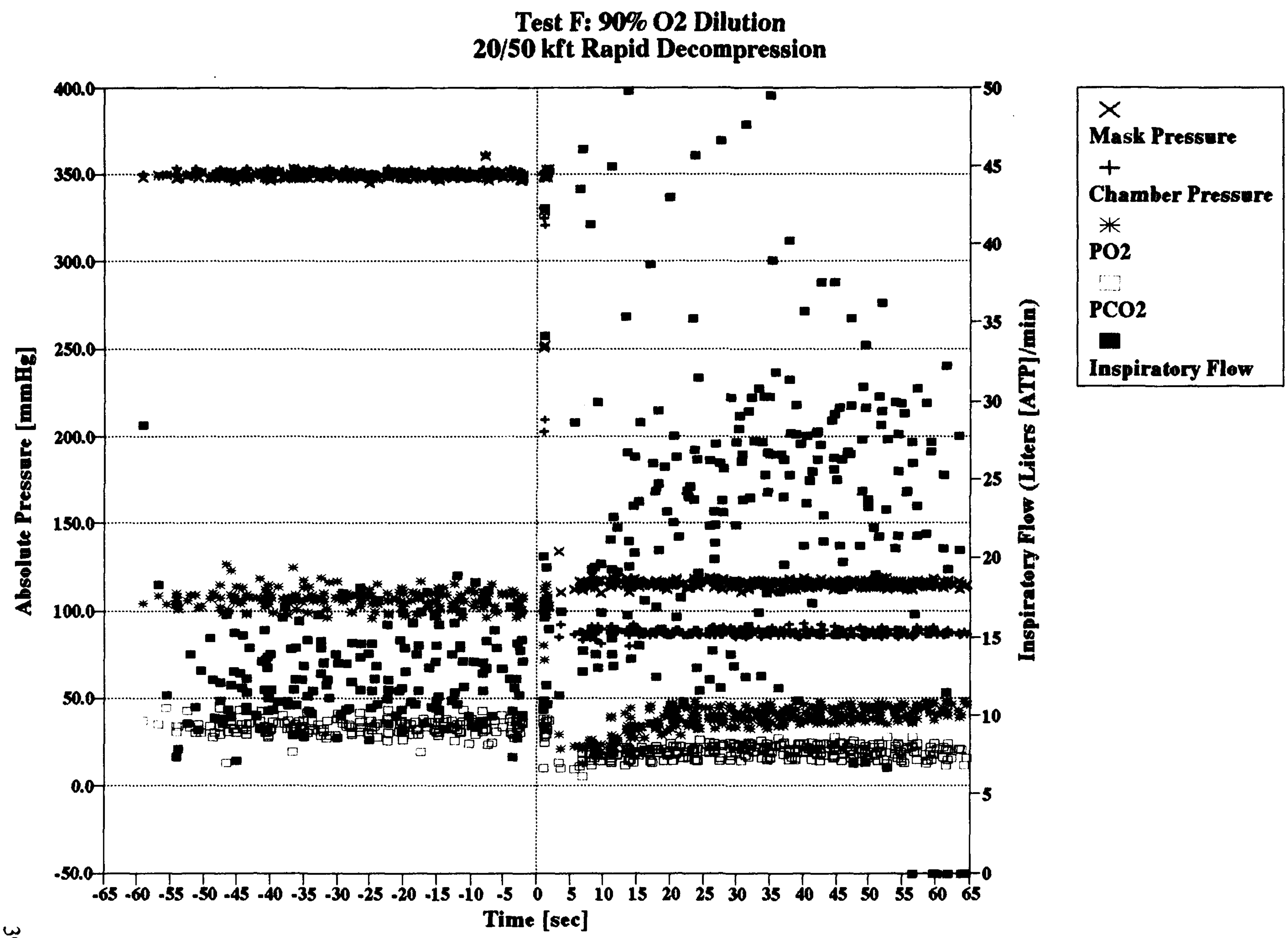


Test D: 93% O2 Dilution
20/50 kft Rapid Decompression



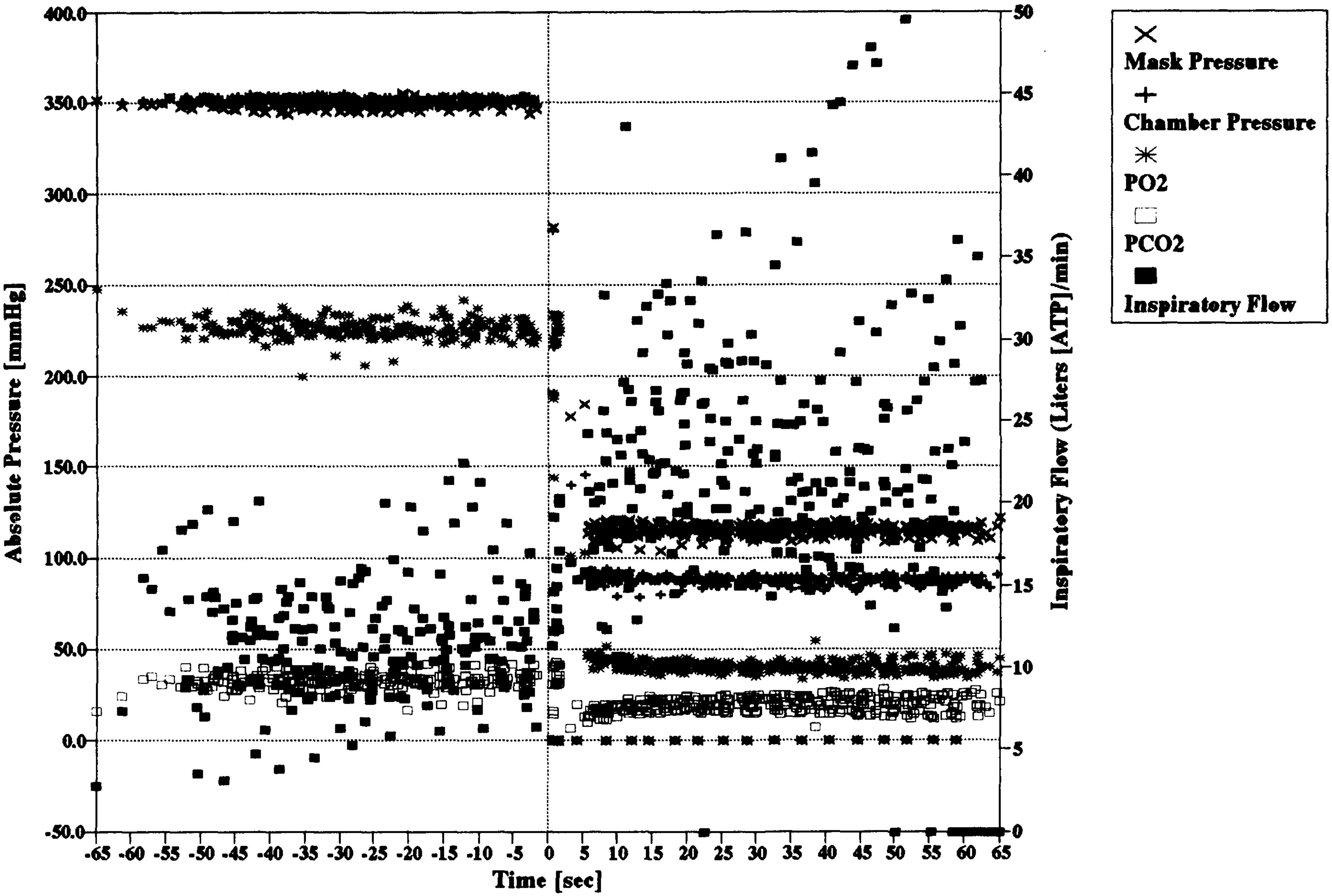
Test E: 90% O2 Non-Dilution
20/50 kft Rapid Decompression



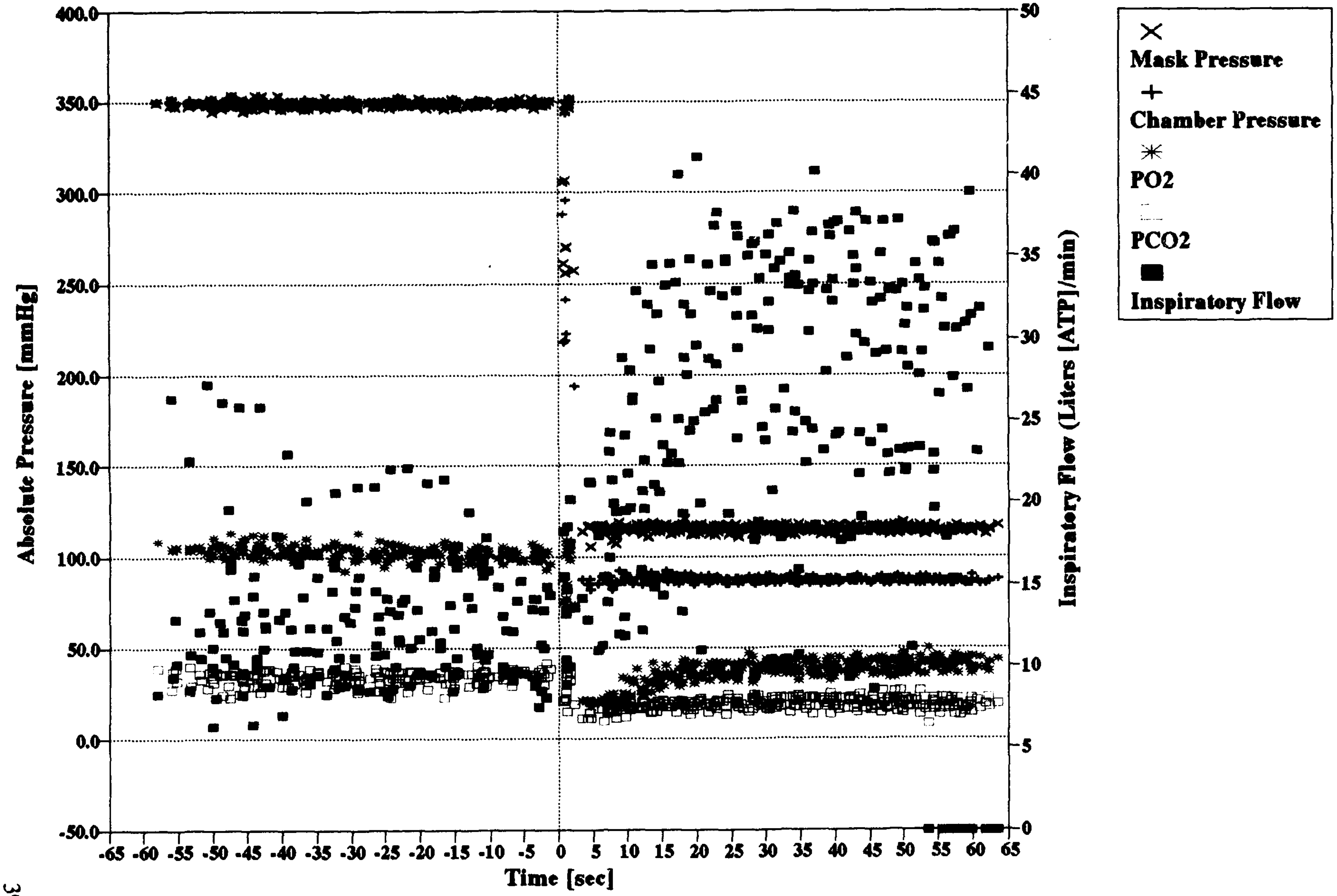


Test F: 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
Inspiratory Flow (Liters [ATP]/min)
50
45
40
35
30
25
20
15
10
5
0
Time [sec]
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Mask Pressure
Chamber Pressure
PO2
PCO2
Inspiratory Flow

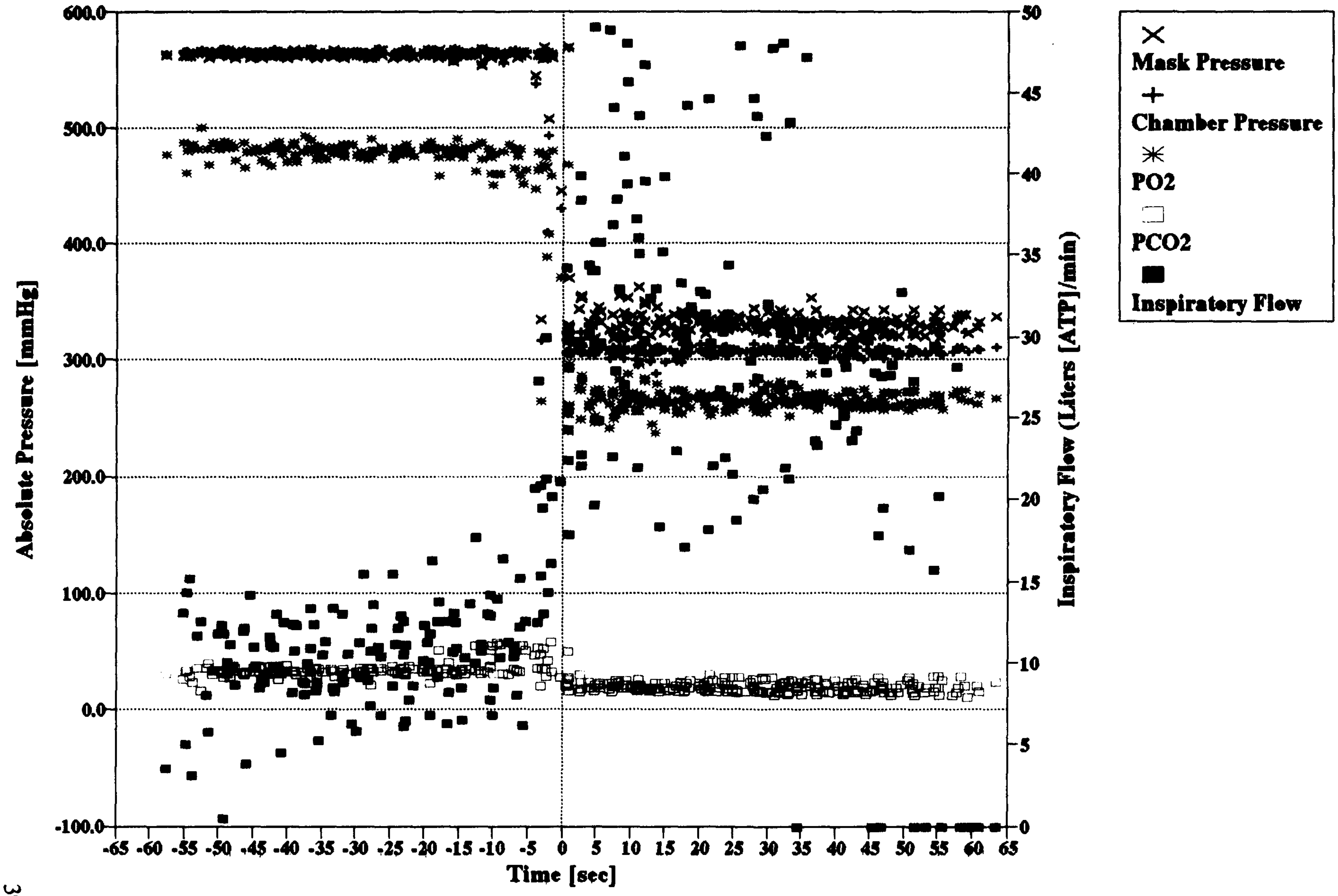
Test G: 85% O2 Non-Dilution
20/50 kft Rapid Decompression



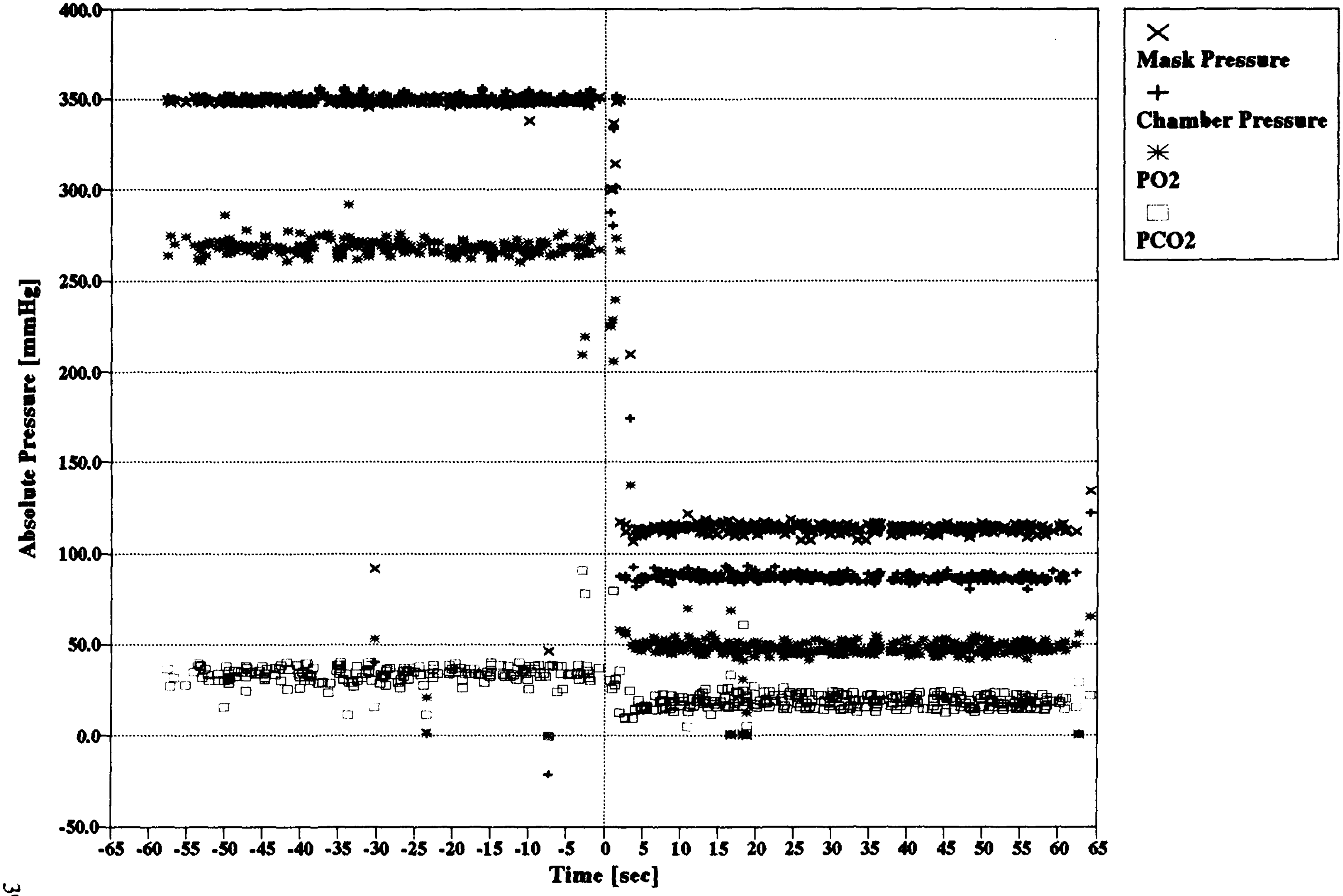
Test H: 85% O2 Dilution
20/50 kft Rapid Decompression



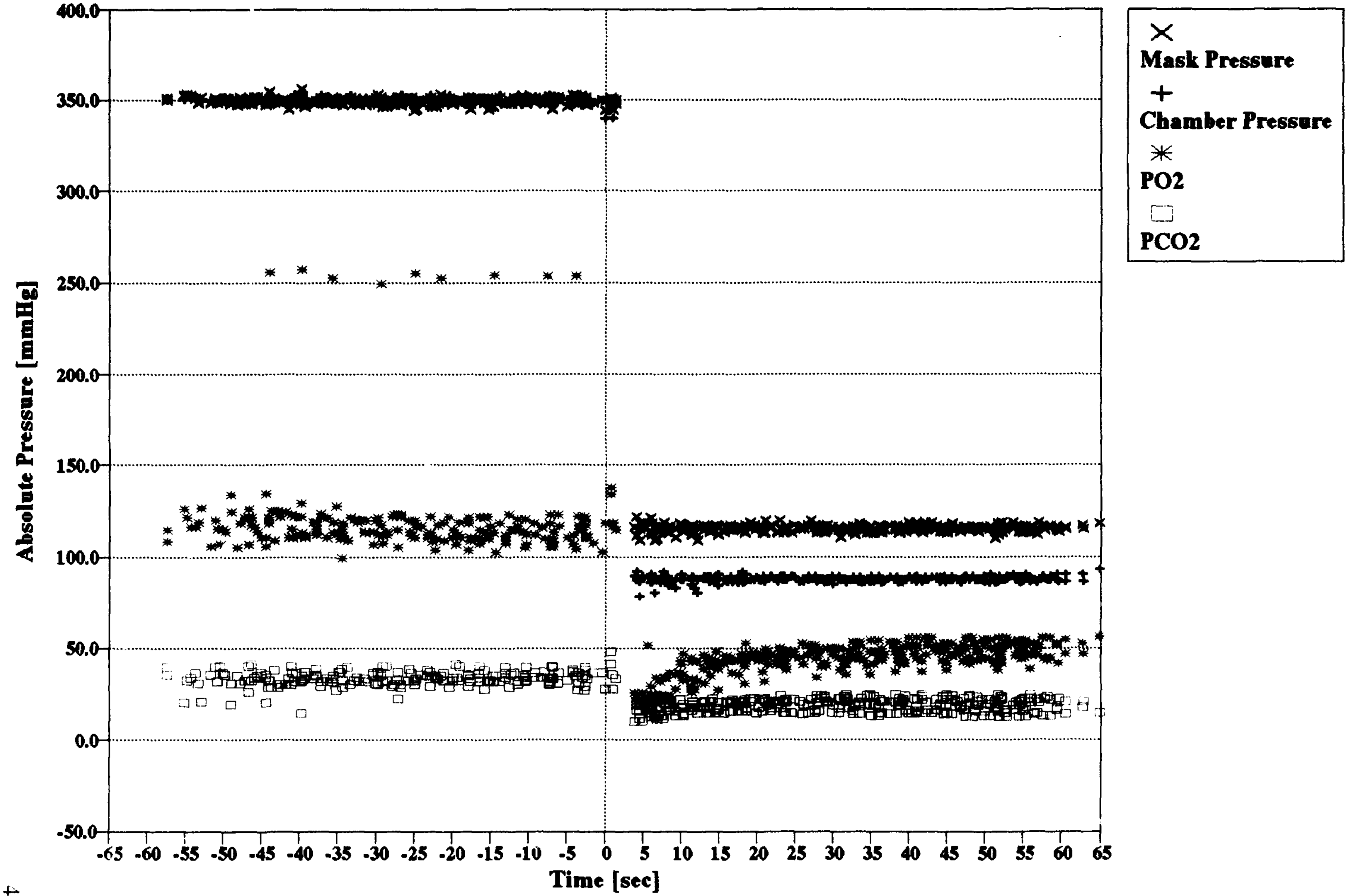
Test EONS2: 100% O2 Non-Dilution
8/20 kft Rapid Decompression



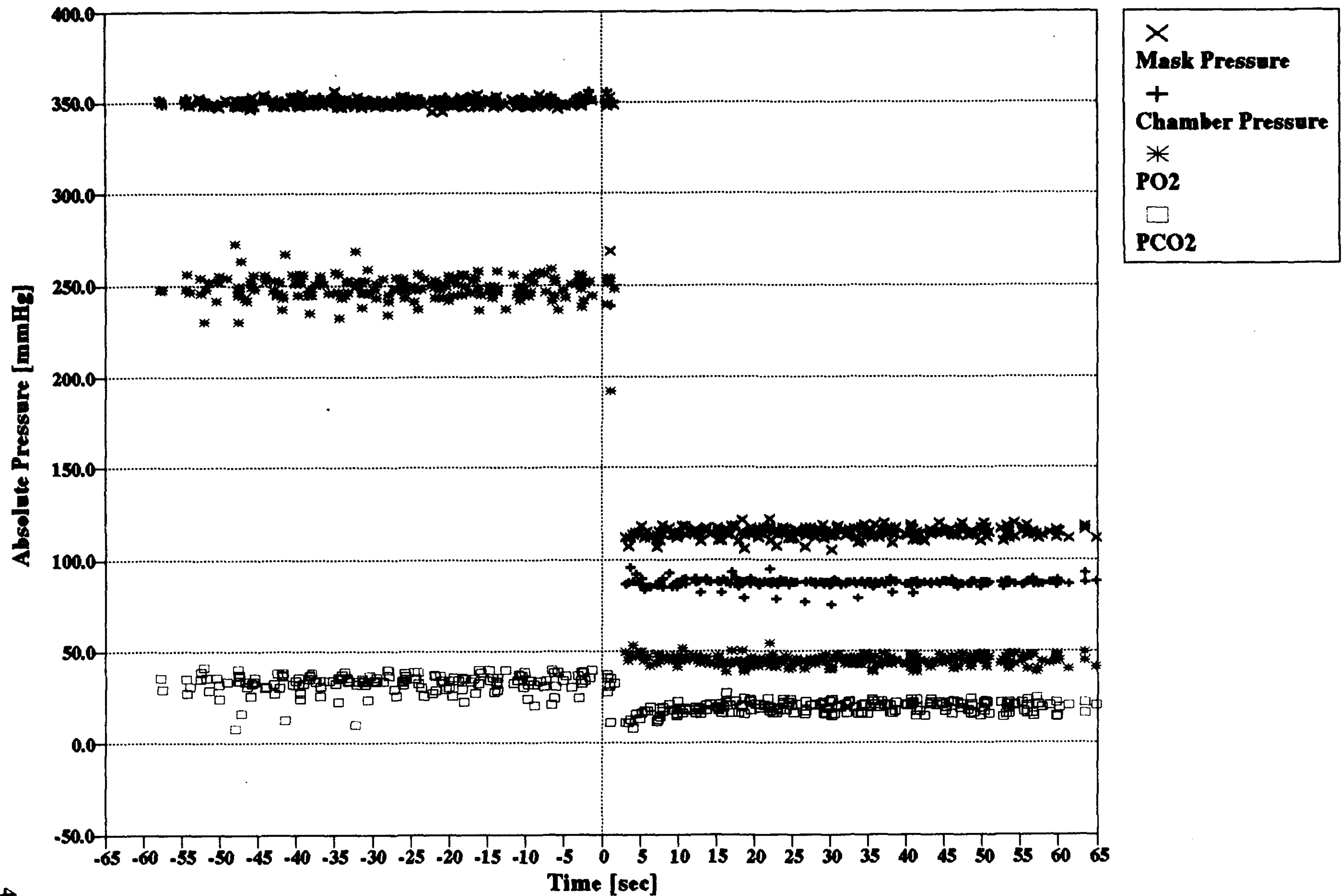
Test A : 100% O2 Non-Dilution
20/50 kft Rapid Decompression



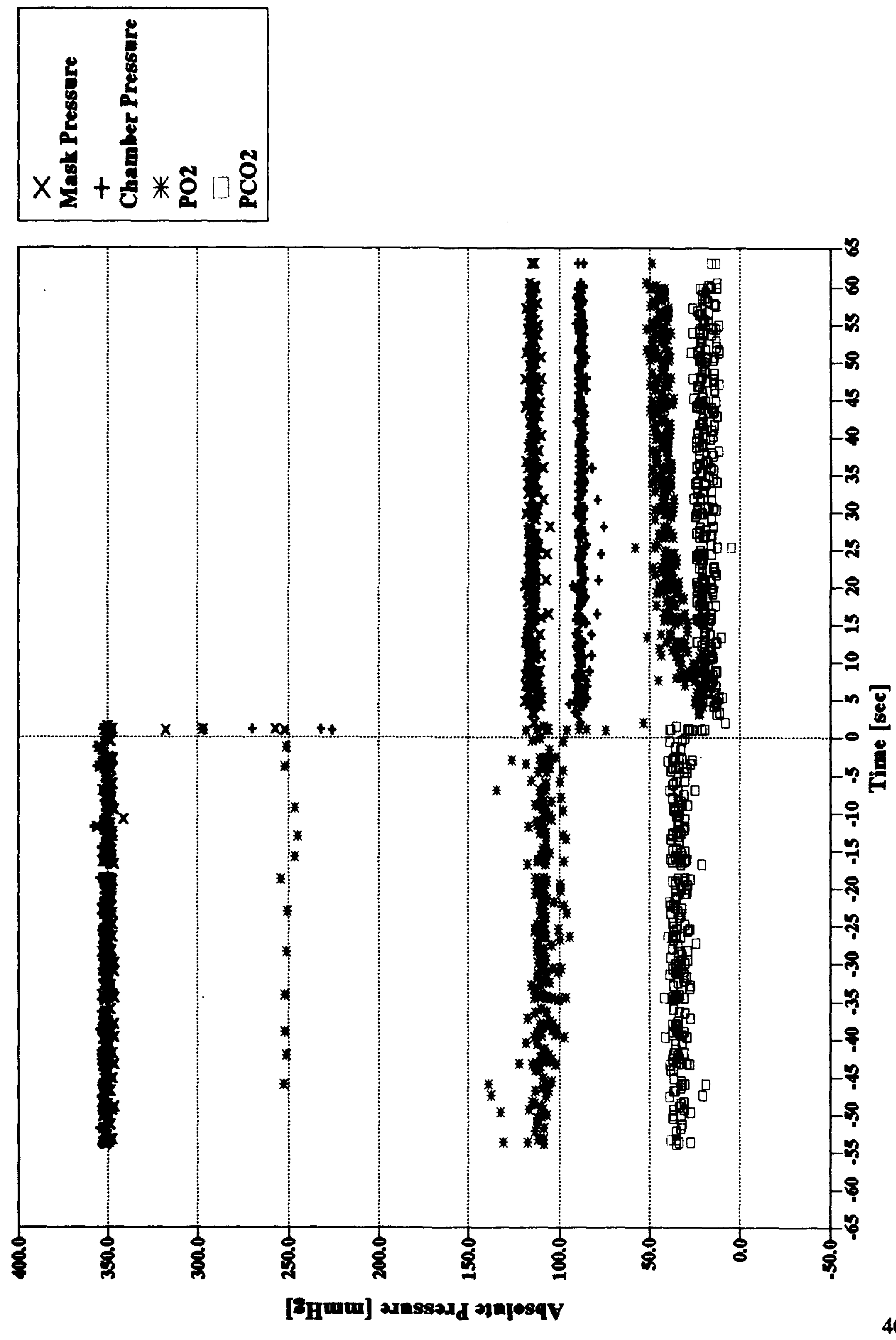
Test B: 100% O2 Dilution
20/50 kft Rapid Decompression



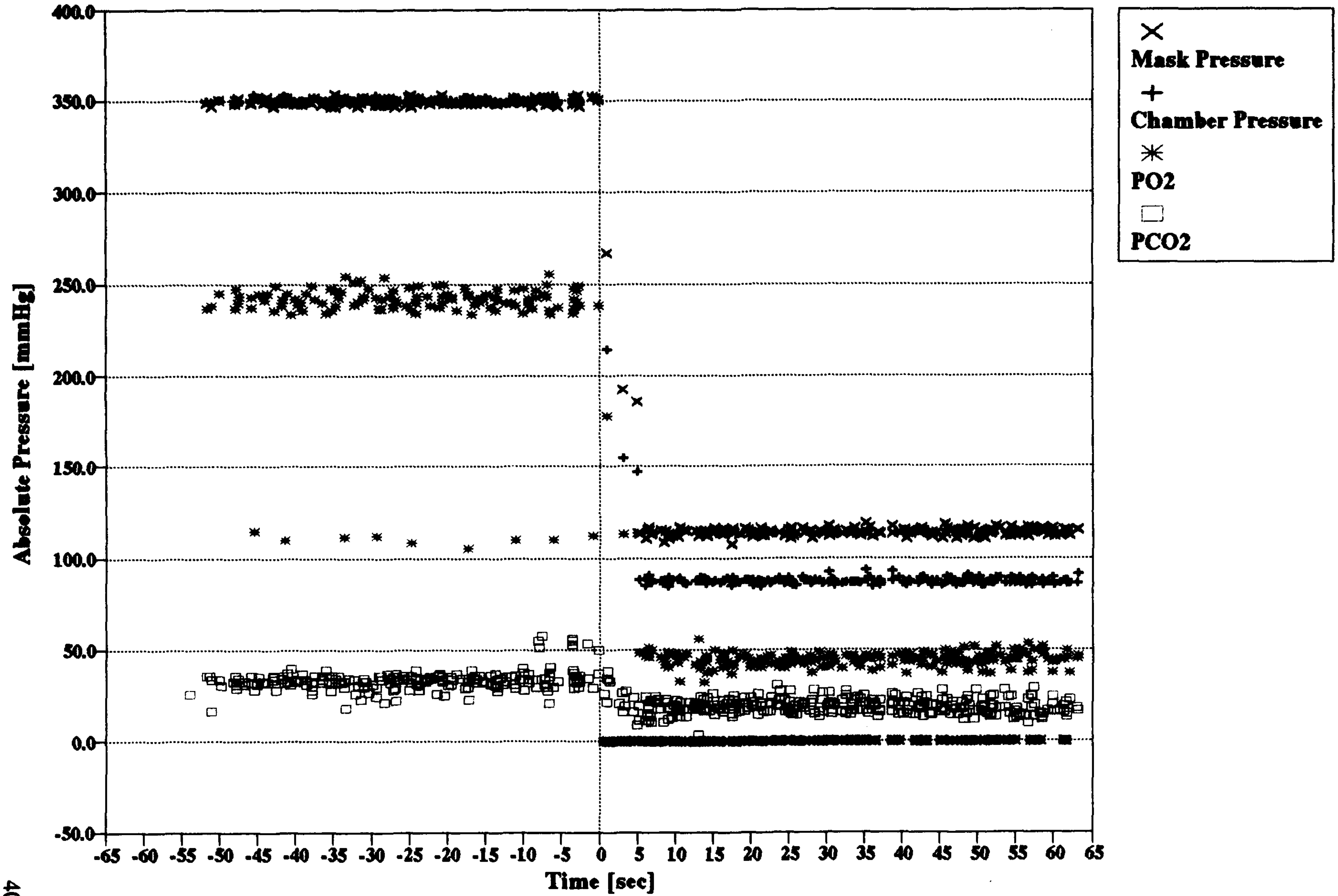
Test C: 93% O2 Non-Dilution
20/50 kft Rapid Decompression

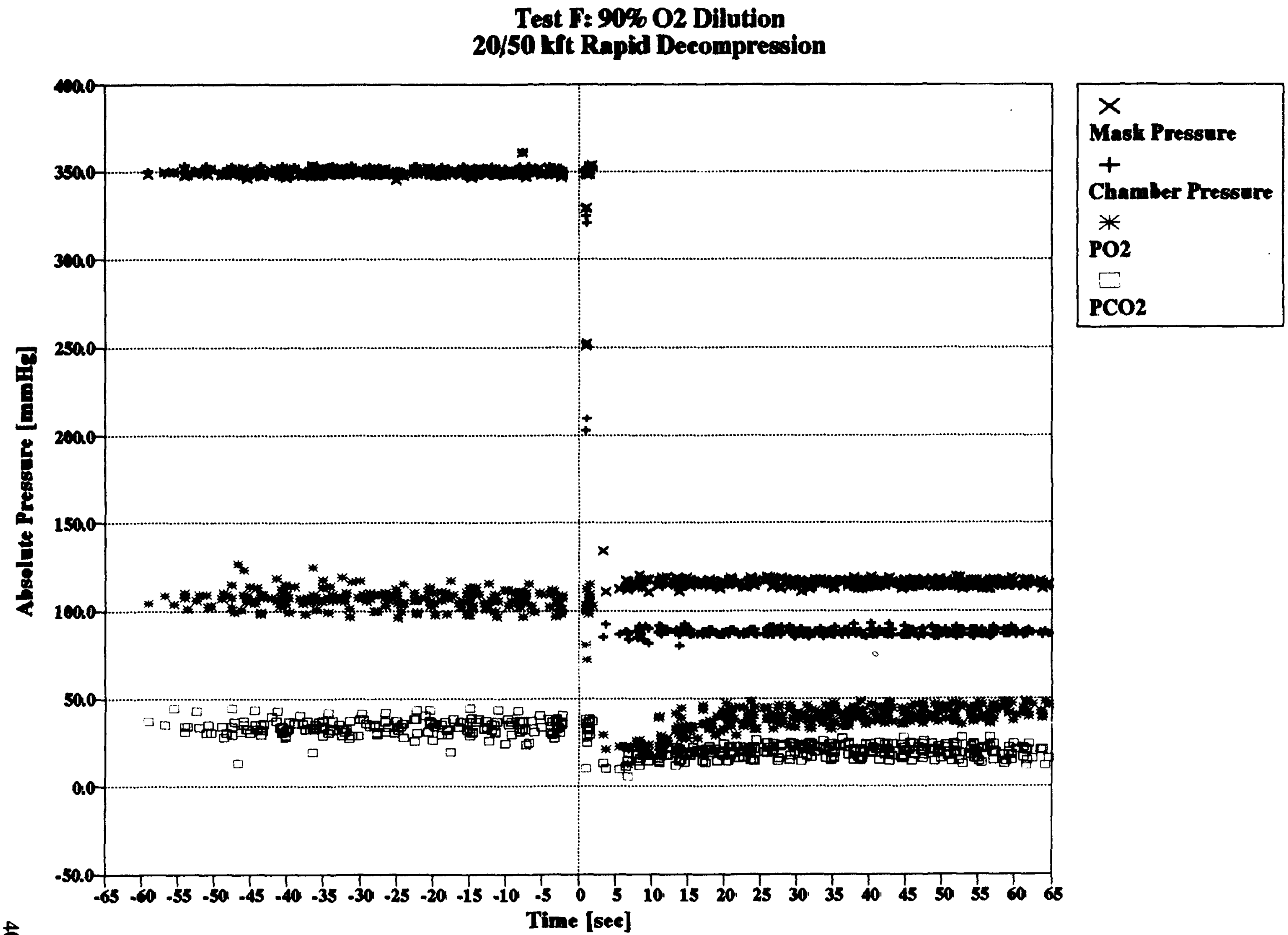


Test D: 93% O2 Dilution
20/50 kft Rapid Decompression

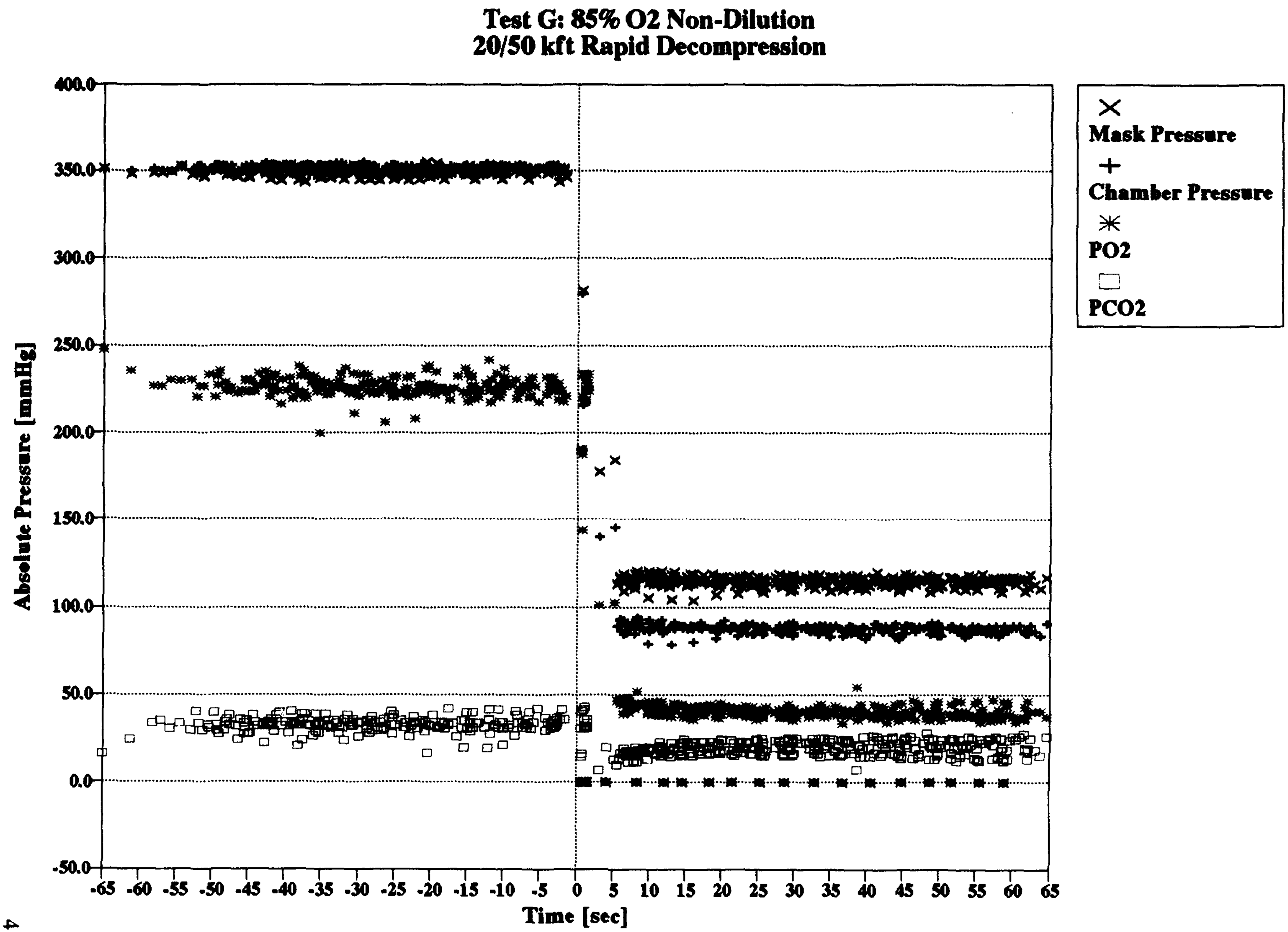


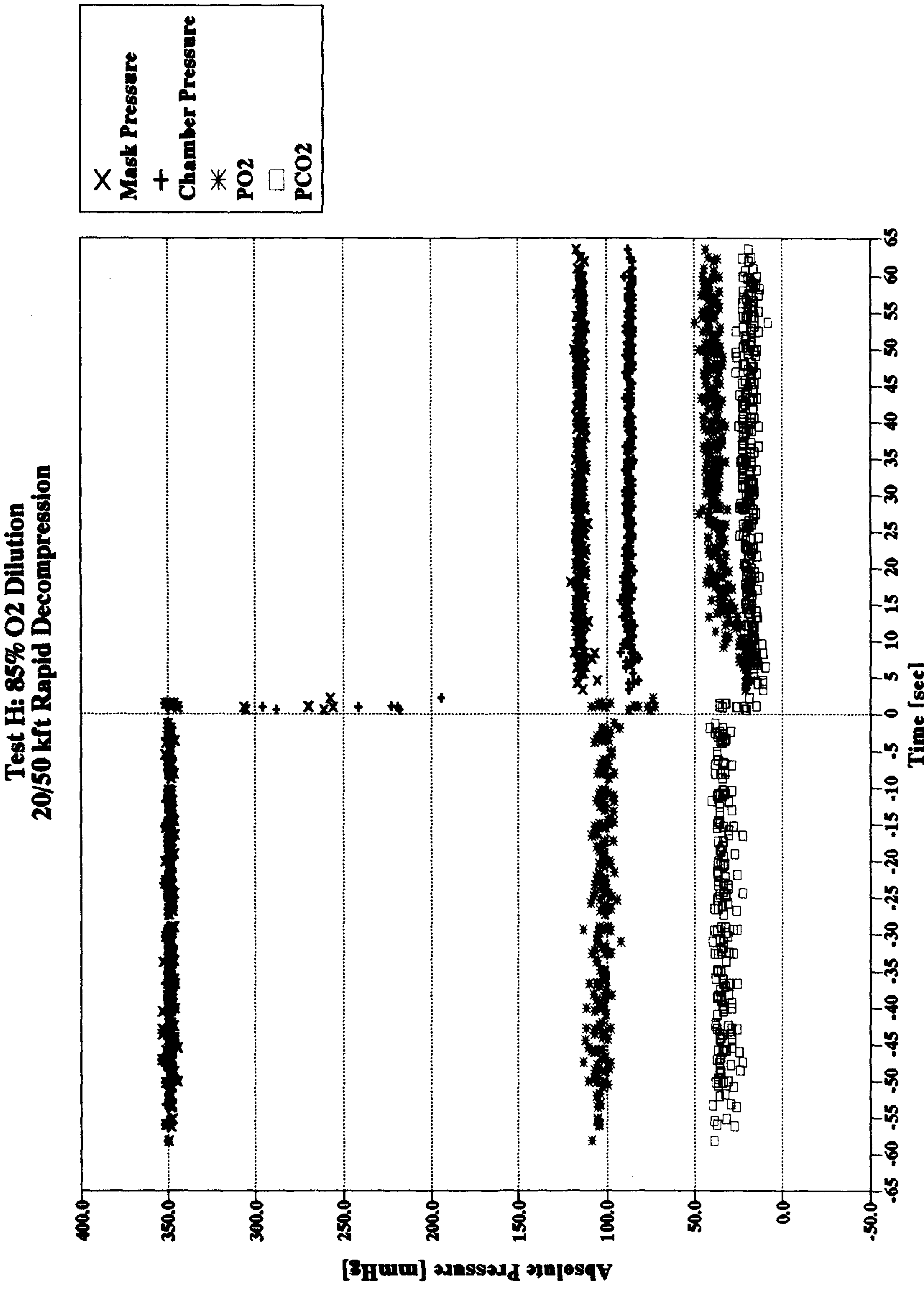
Test E: 90% O2 Non-Dilution
20/50 kft Rapid Decompression



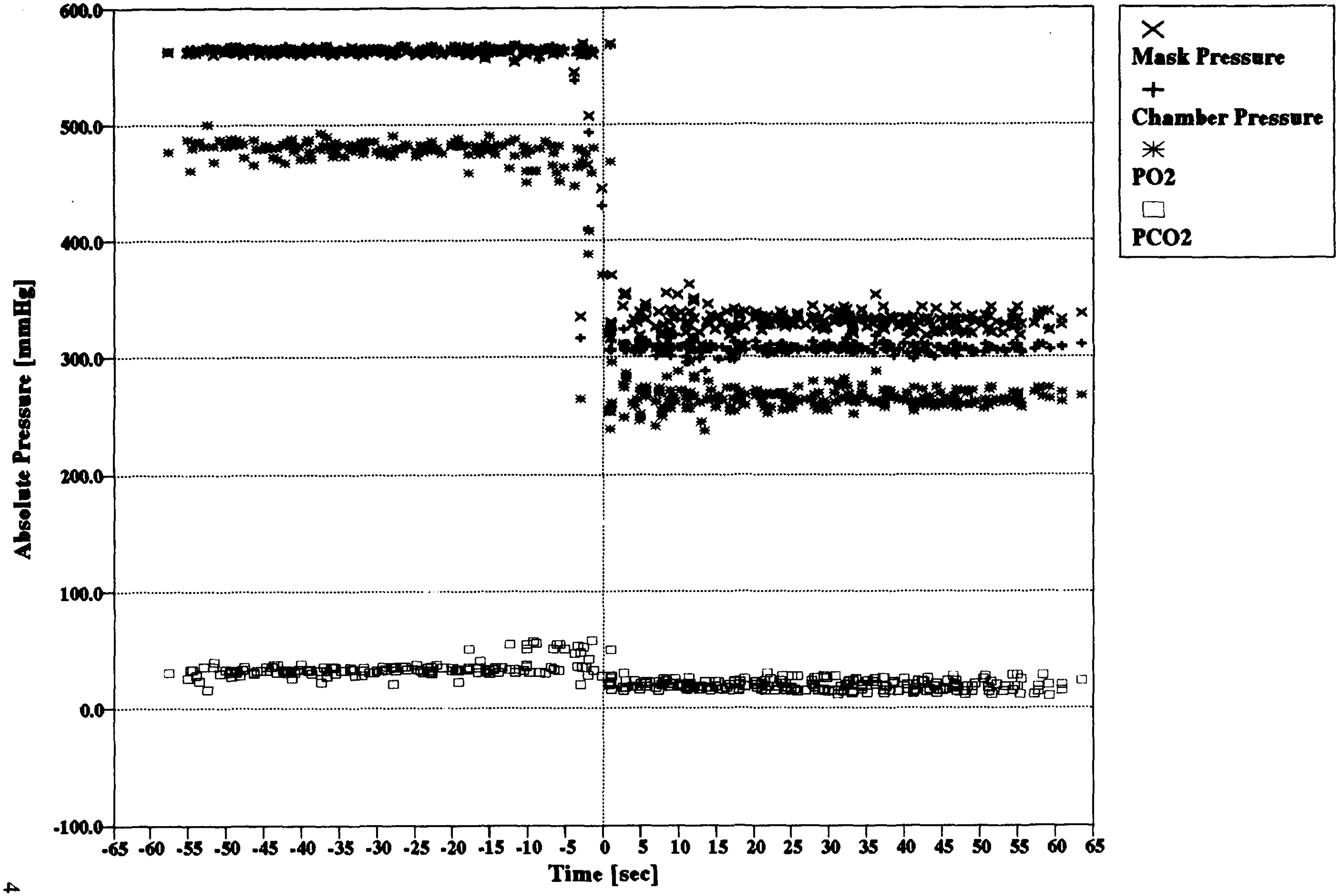


Test F: 90% O2 Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2


Test G: 85% O2 Non-Dilution
20/50 kft Rapid Decompression
Absolute Pressure [mmHg]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
-65 -60 -55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55 60 65
Time [sec]
Mask Pressure
Chamber Pressure
PO2
PCO2


Test H: 85% O2 Dilution
20/50 kft Rapid Decompression
Mask Pressure
Chamber Pressure
PO2
PCO2
Absolute Pressure [mmHg]
Time [sec]
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
-65
-60
-55
-50
-45
-40
-35
-30
-25
-20
-15
-10
-5
0
5
10
15
20
25
30
35
40
45
50
55
60
65

Test EONS2: 100% O2 Non-Dilution
8/20 kft Rapid Decompression



APPENDIX B

Mask/Regulator Model Computer Program

Appendix B
Mask/Regulator Model Computer Program

PROGRAM MASKMODL

```
C...
C...  PROGRAM MASKMODL CALLS: (1) SUBROUTINE INITIAL TO DEFINE THE ODE
C...  INITIAL CONDITIONS, (2) SUBROUTINE RKF45 TO INTEGRATE THE ODES,
C...  AND (3) SUBROUTINE PRINT TO PRINT THE SOLUTION.
C...
C...  THE FOLLOWING CODING IS FOR 500 ODES.  IF MORE ODES ARE TO BE INTE-
C...  GRATED, ALL OF THE 500'S SHOULD BE CHANGED TO THE REQUIRED NUMBER
      IMPLICIT DOUBLE PRECISION (A-H), DOUBLE PRECISION (O-Z)
      INTEGER NI, NO, NEQN, NSTOP, NORUN
      COMMON/T/        T, XT,    NSTOP,     NORUN
     1      /Y/    Y(500)
     2      /F/    F(500)
C...
C...  THE NUMBER OF DIFFERENTIAL EQUATIONS IS IN COMMON/N/ FOR USE IN
C...  SUBROUTINE FCN
      COMMON/N/      NEQN
C...
C...  COMMON AREA TO PROVIDE THE INPUT/OUTPUT UNIT NUMBERS TO OTHER
C...  SUBROUTINES
      COMMON/IO/      NI,        NO
C...
C...  ABSOLUTE DIMENSIONING OF THE ARRAYS REQUIRED BY RKF45
      DOUBLE PRECISION YV(500), WORK(11000)
      INTEGER IWORK(5)
C...
C...  EXTERNAL THE DERIVATIVE ROUTINE CALLED BY RKF45
      EXTERNAL FCN
C...
C...  ARRAY FOR THE TITLE (FIRST LINE OF DATA), CHARACTERS  END OF RUNS
      CHARACTER TITLE(20)*4, ENDRUN(3)*4
C...
C...  DEFINE THE CHARACTERS  END OF RUNS
      DATA ENDRUN/'END ','OF R','UNS '/
C...
C...  DEFINE THE INPUT/OUTPUT UNIT NUMBERS
      NI=5
      NO=6
C...
C...  OPEN INPUT AND OUTPUT FILES
      OPEN(NI,FILE='MASKDAT.DAT')
      OPEN(NO,FILE='OUTPUT.PRN',BLOCKSIZE=2048)
C...
C...  INITIALIZE THE RUN COUNTER
      NORUN=0
C...
C...  BEGIN A RUN
1     NORUN=NORUN+1
C...
C...  INITIALIZE THE RUN TERMINATION VARIABLE
      NSTOP=0
C...
C...  READ THE FIRST LINE OF DATA
C...
      READ(NI,1000,END=999) (TITLE(I), I = 1, 20)
C...
C...  TEST FOR  END OF RUNS  IN THE DATA
C...
      DO 2 I = 1, 3
      IF(TITLE(I) .NE. ENDRUN(I)) GO TO 3
2     CONTINUE
C...
C...  AN END OF RUNS HAS BEEN READ, SO TERMINATE EXECUTION
999   STOP
```

Program MASKMAIN.FOR

```
C...
C...  READ THE SECOND LINE OF DATA
C...
3     READ(NI,*,END=999) TO, TF, TP
C...
C...  READ THE THIRD LINE OF DATA
C...
      READ(NI,*,END=999) NEQN, ERROR
C...
C...  PRINT A DATA SUMMARY
      WRITE(NO,1003)NORUN,(TITLE(I), I = 1, 20),
     1                     TO, TF, TP,
     2                     NEQN, ERROR
      WRITE(*,1003) NORUN, (TITLE(I), I = 1, 20),
     1                     TO, TF, TP,
     2                     NEQN, ERROR

C...
C...  INITIALIZE TIME
      T = TO
C...
C...  SET THE INITIAL CONDITIONS
      CALL INITIAL
C...
C...  SET THE INITIAL DERIVATIVES (FOR POSSIBLE PRINTING)
      CALL DERV
C...
C...  PRINT THE INITIAL CONDITIONS
      CALL PRINT(NI, NO)
C...
C...  SET THE INITIAL CONDITIONS FOR SUBROUTINE RKF45
      TV = TO
      DO 5 I = 1, NEQN
      YV(I) = Y(I)
5     CONTINUE
C...
C...  SET THE PARAMETERS FOR SUBROUTINE RKF45
C...
      RELERR = ERROR
      ABSERR = ERROR
      IFLAG = 1
      TOUT = TO + TP
C...
C...  CALL SUBROUTINE RKF45 TO START THE SOLUTION FROM THE INITIAL
C...  CONDITION (IFLAG = 1) OR COMPUTE THE SOLUTION TO THE NEXT PRINT
C...  POINT (IFLAG = 2)
4     CALL RKF45(FCN,NEQN,YV,TV,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
C...
C...  PRINT THE SOLUTION AT THE NEXT PRINT POINT
      T=TV
      TOUT = TV + TP
      PRINT *,"Time = ", T
      DO 6 I = 1, NEQN
      Y(I) = YV(I)
6     CONTINUE
      CALL DERV
      CALL PRINT(NI,NO)
      IF(IFLAG .EQ. 4 .OR. IFLAG .EQ. 7) IFLAG = 2
C...
C...  TEST FOR AN ERROR CONDITION
      IF(IFLAG .NE. 2) THEN
C...
C...     PRINT A MESSAGE INDICATING AN ERROR CONDITION
         WRITE(NO,1004) IFLAG
C...
```

Program MASKMAIN.FOR

```
C...      GO ON TO THE NEXT RUN
          GO TO 1
       END IF
C...
C...   CHECK FOR A RUN TERMINATION
       IF(NSTOP .NE. 0) GO TO 1
C...
C...   CHECK FOR THE END OF THE RUN
C...
       IF(TV .LT. (TF - 0.5D0*TP)) GO TO 4
C...
C...   THE CURRENT RUN IS COMPLETE, SO GO ON TO THE NEXT RUN
       GO TO 1
C...
C...   ****************************************************************
C...
C...   FORMATS
C...
1000   FORMAT(20A4)
1001   FORMAT(3E10.0)
1002   FORMAT(I5,20X,E10.0)
1003   FORMAT(1H1,
     1 ' RUN NO. - ',I3,2X,20A4,//,
     2 ' INITIAL T - ',E10.3,//,
     3 '   FINAL T - ',E10.3,//,
     4 '   PRINT T - ',E10.3,//,
     5 ' NUMBER OF DIFFERENTIAL EQUATIONS - ',I5,//,
     6 ' MAXIMUM INTEGRATION ERROR - ',E10.3,//,
     7 1H1)
1004   FORMAT(1H ,//,' IFLAG = ',I3,//,
     1 ' INDICATING AN INTEGRATION ERROR, SO THE CURRENT RUN'     ,/,
     2 ' IS TERMINATED.  PLEASE REFER TO THE DOCUMENTATION FOR'   ,/,
     3 ' SUBROUTINE',//,25X,'RKF45',//,
     4 ' FOR AN EXPLANATION OF THESE ERROR INDICATORS'            )
       END
       SUBROUTINE FCN(TV,YV,YDOT)
C...
C...   SUBROUTINE FCN IS AN INTERFACE ROUTINE BETWEEN SUBROUTINES RKF45
C...   AND DERV
C...
C...   NOTE THAT THE SIZE OF ARRAYS Y AND F IN THE FOLLOWING COMMON AREA
C...   IS ACTUALLY SET BY THE CORRESPONDING COMMON STATEMENT IN MAIN
C...   PROGRAM HEADHIT
       IMPLICIT DOUBLE PRECISION (A-H), DOUBLE PRECISION (O-Z)
       INTEGER NEQN, NSTOP, NORUN
       COMMON/T/          T,  XT,    NSTOP,     NORUN
     1       /Y/       Y(500)
     2       /F/       F(500)
C...
C...   THE NUMBER OF DIFFERENTIAL EQUATIONS IS AVAILABLE THROUGH COMMON
C...   /N/
C...
       COMMON/N/       NEQN
C...
C...   ABSOLUTE DIMENSION THE DEPENDENT VARIABLE, DERIVATIVE VECTORS
       DOUBLE PRECISION YV(500), YDOT(500)
C...
C...   TRANSFER THE INDEPENDENT VARIABLE, DEPENDENT VARIABLE VECTOR
C...   FOR USE IN SUBROUTINE DERV
C...
       T = TV
       DO 1 I = 1, NEQN
       Y(I) = YV(I)
1      CONTINUE
C...
```

Program MASKMAIN.FOR

```
C...  EVALUATE THE DERIVATIVE VECTOR
C...
      CALL DERV
C...
C...  TRANSFER THE DERIVATIVE VECTOR FOR USE BY SUBROUTINE RKF45
C...
      DO 2 I = 1, NEQN
      YDOT(I) = F(I)
2     CONTINUE
      RETURN
      END
```

Program MASKSUBS.FOR

```
C...  DECK MASKSUBS.FOR - SUBROUTINES REQUIRED TO IMPLEMENT A MODEL OF AN
C...  AVIATOR'S ORONASAL MASK.
C...
      SUBROUTINE INITIAL
C...
C...   Subroutine Initial sets initial conditions and defines constants
C...   passed to other modules in COMMON.  It is only called once.
C...
C...  DETAILED EXPLANATION OF THE EQUATIONS.
C...
C...
C...  NOTE:  The subscript notation indicated partial derivative WRT the
C...         subscript  E.G.  Xt ==> The first partial derivative of X
C...         WRT time.
C...
C...  This model estimates the flow and pressure within the mask hose and
C...  the oronasal cavity of the aviator's breathing mask during the
C...  breathing cycle.  The forcing function for the model is the volume
C...  change generated by the lung during breathing and pressure changes.
C...
C...     The present coding models the lung volume change by the following
C...  sinusoidal function.
C...
C...      V(t) = VLO - Qmax/w*(COS(wt) - 1), so that the lung flow is
C...
C...      Q(t) = Qmax*SIN(wt).
C...
C...      The airway resistances are modeled by simple quadratic fits to
C...  physiologic data.  The resistance of the bronchi and trachea are
C...  included in the lung model, but the resistance of the oronasal cavity
C...  is included in the mask model for convenience.  The oronasal resis-
C...  tances are accounted for separately for the nose and mouth by the
C...  following model.
C...
C...      deltaP =  K1*Q + K2*Q^2
C...
C...      The last term in the above accounts for inertance.
C...
C...      For the mouth,
C...
C...      K1 = 2.4 cm-H2O*sec/liter and K2 = 0.3 cm-H2O*sec^2/liter^2.
C...
C...      For the nose there are seperate K2's for inspiration and
C...  expiration.
C...
C...      K1 = 3.0 cm-H2O*sec/liter and
C...
C...      Inspiratory K2 = 3.0 cm-H2O*sec^2/liter^2.
C...
C...      Expiratory  K2 = 4.0 cm-H2O*sec^2/liter^2.
C...
C...      The overall pressure drop produced by the oronasal cavity
C...  is modelled by assigning the relative fraction of the total breathing
C...  flow to the mouth and nose respectively.  The individual drops are
C...  weighted by the flow fractions.  Thus,
C...
C...       Fm + Fn = 1.0, and deltaP = Fm*deltaPm + Fn*deltaPn.
C...
C...       The mask is modelled as a dead space and two variable area
C...  orifices through which inspiratory and expiratory flows separately
C...  pass.  The inspiratory valve model estimates the flow between
C...  the mask supply hose and the oronasal cavity during inspiration.
C...  The expiratory valve model relates the flow and pressure drop between the
C...  mask cavity and the external ambient atmosphere during expiratory flow.
C...  Both valve models are based on empirical data collected on the RAF
```

Program MASKSUBS.FOR

```
C...  P/Q mask, which meets the ASCC flow resistance standard.  Other valve
C...  models can be substituted.  The general form of the model is
C...
C...     Q = F(A,deltaP),
C...
C...  where A is the mask valve area and Q is the instantaneous
C...  flow through the valve.  The valve area, A, is in turn a function
C...  of the pressure drop, deltaP, across the valve.  It is solved by
C...  estimating the area of the valve from the pressure drop across the
C...  valve and then computing the flow through the valve using ideal
C...  orifice equations.  Mask leaks can be modelled by parallel flow
C...  paths to ambient, but none are included in this version.  Expiratory
C...  valve compensation is simulated by adding any positive difference
C...  between mask hose pressure and ambient pressure to the expiratory
C...  valve down stream pressure which is normally ambient pressure.
C...
C...  Mask hose and connector losses are modelled as simple tubes with
C...  flow-pressure drop relationships based on empirical data.  The
C...  regulator outlet flow-pressure relationship is based on curve fits
C...  to empirical data.
C...
C...
C...
C...  ODE COMMON
C...
C...   /Y/ time variables
C...   /F/ time derivatives of variables
C...   /S/ spatial derivatives of variables
C...   /R/ & /I/ real and integer parameters required to define constants and
C...           define the spatial integration grid.
C...
      PARAMETER (NDE=3)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DOUBLE PRECISION  KG, KP1, KP1OK, KOKM1, KOKP1
      INTEGER NSTOP, NORUN, IP, NEQN
      COMMON/T/          T, XT,    NSTOP,     NORUN
     1     /Y/ VOL(NDE), FLOW(NDE), PRESS(NDE),          !
C...
C...           VOL(1)  [=]  Cumulative volume flow from regulator  [m^3]
C...           VOL(2)  [=]  Cumulative volume flow into mask       [m^3]
C...           VOL(3)  [=]  Cumulative Respired volume (Insp - Exp)[m^3]
C...
C...           FLOW(1) [=]  Instantaneous flow from regulator [m^3/sec]
C...           FLOW(2) [=]  Instantaneous mask flow       [m^3/sec]
C...           FLOW(3) [=]  Instantaneous Oro-nasal flow  [m^3/sec]
C...
C...           PRESS(1) [=]  Regulator Outlet Pressure  [Pa]
C...           PRESS(2) [=]  Mask/Oronasal Cavity Press.[Pa]
C...           PRESS(3) [=]  Intra Pulmonary Pressure   [Pa]
C...
     2     /F/ DVDT(NDE), DFDT(NDE), DPDT(NDE),
C...
C...       Derivatives of Volumes and Pressures
C...
     3     /S/ DUMMY                  !   Spatial Derivatives if needed
     4     /R/ PI, PO, VO, TO, RUNIV,  ! Pi = 3.1412..., Std P,V,T& R
     *         PA2MM,
     *         KG, KP1, KOKM1, KP1OK, KOKP1,! K, K/(K-1), (K+1)/K, 1/K, 2/K, etc
     *         TWOOK, TMKOK, TWOOKP1,  ! Adiabatic constant and
     *         TWOOKM1, ONEOK,         ! derived constants
     *         CD,                     ! Orifice Discharge Coefficient
     *         TBODY, TAMB, PAMB,      ! Temperatures - Body & Ambient
     *         GMW, GMWS, GASK, GASKS, ! Gas parameters MW & Specif Heat ratios
     *         GAM, OMEGA, TINSP, FRN, ! Breathing flow parameters
     *         TR, TPAUSE, RR, VTIDAL, ! Breathing flow parameters
```

Program MASKSUBS.FOR

```
     *          VFRCO, FRC, PPWBODY,    ! Initial, Current FRC, ppH2O Body
     *          AM(2), ANI(3), ANE(3),  ! Airway parameters
     *          VAO, VMAO, CINERT, DPAW,! Airway parameters
     *          AWI, DVL,               ! Airway inertance
     *          PREGO, PREF, Gdot1,     ! Regulator Outlet Pressure
     *          VDSMSK, AMIV, AMEV,     ! Mask parameters
     *          QIV, QEV,               ! Mask valve flow indicators
     *          VOL1, VOL2, VOL3, VOL4, ! Volumes
     *          AWH, AWO, AWN, AWAr,    ! Atomic wts
     *          GMN2, GMO2, GMCO2,      ! Molecular wts
     *          GMH2O, GMWAIR,          !    "     "
     *          FINSP(4), FEXP(4)       !Gas Fractions, O2,N2+Ar,CO2,H2O
     5      /I/ IP , IRELF              ! Print Counter
     6      /N/ NEQN
C...
C...  Set Constants employed in simulation
C...
C...  Pi = 3.14159...
C...  T0 = Freezing point of water at 1 atmosphere
C...  V0 = Volume occupied by 1 kg-mol of ideal gas at 1 atm and Tzero
C...  P0 = Standard atmospheric pressure in Pascals [N/m^2]
C...
      Pi = DACOS(-1.D0)
      T0 = 273.1D0            ! deg K
      V0 = 22.4097D0          ! m^3/kg-mol
      P0 = 1.01325D5          ! Pa
      GASK = 1.4D0            ! k = Specific Heat Ratio Cp/Cv
      CD = 0.68               ! Orifice coefficient
      FM  = 0.75D0            ! Fraction mouth breathing
      VAO = 150.D-6           ! Respiratory Anatomical Dead Space (m^3)
      VDSMSK = 150.D-6        ! Oronasal Mask Dead Space (m^3)
      VFRCO = 2.5D-3          ! Resting Functional Residual Capacity [m^3]
      DVL = 2.2502D-7         ! Rate of Lung Volume Increase per Pascal of PPB
      GMWH = 1.008D0          ! Atomic Wt Hydrogen
      AWN = 14.01D0           ! AW Nitrogen
      AWO = 16.D0             ! AW Oxygen
      AWC = 12.01D0           ! AW Carbon
      AWAr = 39.94D0          ! AW Argon
C...
C...  DERIVED CONSTANTS
C...
C...  PV = RT  Universal Gas Law
C...
C...  RUNIV = P0*V0/T0  Universal Gas Constant
C...
      RUNIV = P0*V0/T0
      KP1 = GASK + 1.D0               ! k + 1
      KOKM1 = GASK/(GASK - 1.D0)      ! k/(k - 1)
      KP1OK = KP1/GASK                ! (k + 1)/k
      KOKP1 = GASK/KP1                ! k/(k + 1)
      TWOOK = 2.D0/GASK               ! 2/k
      TMKOK = (2.D0 - GASK)/GASK      ! (2 - k)/k
      TWOOKP1 = 2.D0/KP1              ! 2/(k + 1)
      TWOOKM1 = 2.D0/(GASK - 1.D0)    ! 2/(k - 1)
      ONEOK   = 1.D0/GASK             ! 1/k
      TAMB    = T0 + 25.D0            ! Ambient Temperature
      TBODY =  T0 + 37.D0             ! Body Temperature
      PA2MM = 760.D0/1.01325D5        ! Conversion Factor Pa to mmHg
      PAMB  = 1.01325D5
      PREF  = PAMB
      GMH2O = AWH*2.D0 + AWO          ! MW Water
      GMCO2 = AWC + 2.D0*AWO          ! MW Carbon Dioxide
      GMO2  = 2.D0*AWO                ! MW N2
      GMN2  = 2.D0*AWN                ! MW O2
      GMWAIR = 0.2095D0*GMO2
```

```
     *       + 0.7808D0*GMN2
     *       + 0.0093D0*AMAr          ! MW Dry Air (Neglect 0.04% CO2)
      AWR = ((VAO + VDSMSK)*3.D0/4.D0/Pi)**(1.D0/3.D0) ! Airway radius
      AWL = 2.D0*AWR        ! Characteristic length of airway
      AWA = Pi*AWR*AWR ! Characteristic airway area
      AWI = AWL/AWA         ! Airway Inertance (m^-1)
C...
C...
C...  INITIAL CONDITIONS (T = 0)
C...
C...   RESPIRATION RATE (Breaths/min)
C...
      RR = 20.D0/60.D0      !  Breaths per second
      TR = 1.D0/RR          !  Respiratory period
      TINSP = TR/2.D0       !  Inspiratory period
      TPAUSE = 0.1D0*TR     !  Interbreath pause duration
C...
C...   VDOTE = MINUTE VOLUME (M^3/SEC)
C...
      VTIDAL = 1.5D-3       !  Tidal Volume = 1.5 liter/breath
      VDOTE = RR*VTIDAL     !  VdotE in m^3/sec
      QAM = VDOTE*Pi        !  Peak Flow from sinusoidal flow profile
      OMEGA = 2.D0*Pi*RR    !  RR in radians/sec
      FRM = 0.75D0          !  Fraction Mouth Breathing
      FINSP(1) = 0.5D0      !  Fr O2 inspired
      FINSP(2) = 0.5D0      !  Fr N2 (Inert) inspired
      FINSP(3) = 0.0D0      !  Fr CO2 inspired
      FINSP(4) = 0.0D0      !  Fr H2O inspired
      FEXP(1) = 0.17D0      !  Fr O2 expired
      FEXP(3) = 0.04D0      !  Fr CO2 expired
      FEXP(4) = 0.062D0     !  Fr H2O expired
      FEXP(2) = 1.D0 - (FEXP(1) + FEXP(3) + FEXP(4))  ! Fr N2 (Inert) expired
      VOL(1) = Pi*((2.54D-2)/2.D0)**2.D0*2.D0 ! 2 m of 1 in ID hose
      VOL1 = VOL(1)           !  Parameterize initial volumes
      VOL(2) = VDSMSK + VAO !  Include mask cavity and anatomical dead space
      VOL2 = VOL(2)
      VOL(3) = VFRCO
      VOL3 = VOL(3)
      PREGO = PAMB            ! Initialize regulator outlet pressure
      AMIV = 0.D0
      AMEV = 0.D0
      QIV = 0.D0
      QEV = 0.D0
      PPWBODY = 47.D0/PA2MM ! Convert saturation pressure to Pascals
      i = 1
      DO WHILE (i .LE. NDE)! Initialize flows, pressures, and derivatives
         PRESS(i) = PAMB
         FLOW(i)  = 0.D0
         DFDT(i)  = 0.D0
         DVDT(i)  = 0.D0
         DPDT(i)  = 0.D0
         i = i + 1
      END DO
C...
C...  Compute Starting values for derivatives by calling DERV
C...
      CALL DERV
      IP = 0        !  Initialize print flag
      IRELF = 0  !  Initialize RD flag
      RETURN
      END
C...
      SUBROUTINE DERV
C...
C...  DERV CALCULATES THE TIME DERIVATIVES TO BE INTEGRATED BY RKF45
```

Program MASKSUBS.FOR

```
C...
C...  ODE COMMON
C...
C...   /Y/ time variables
C...   /F/ time derivatives of variables
C...   /S/ spatial derivatives of variables
C...   /R/ & /I/ real and integer parameters required to define constants and
C...         define the spatial integration grid.
C...
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DOUBLE PRECISION  KG, KP1, KP1OK, KOKM1, KOKP1
      DOUBLE PRECISION  DPMIE, DPMI, DPME
      INTEGER NSTOP, NORUN, NDE, IP, NEQN
      PARAMETER (NDE=3)
      COMMON/T/        T, XT,    NSTOP,     NORUN
     1      /Y/ VOL(NDE), FLOW(NDE), PRESS(NDE),       !
C...
C...          VOL(1)  [=]  Cumulative volume flow from regulator
C...          VOL(2)  [=]  Cumulative volume flow into mask
C...          VOL(3)  [=]  Cumulative Respired volume (Insp - Exp)
C...
C...          FLOW(1) [=]  Instantaneous flow from regulator [m^3/sec]
C...          FLOW(2) [=]  Instantaneous mask flow       [m^3/sec]
C...          FLOW(3) [=]  Instantaneous Oro-nasal flow  [m^3/sec]
C...
C...          PRESS(1) [=]  Regulator Outlet Pressure
C...          PRESS(2) [=]  Mask hose pressure
C...          PRESS(3) [=]  Mask cavity Pressure = Intraoral pressure
C...
     2      /F/ DVDT(NDE), DFDT(NDE), DPDT(NDE),
C...
C...      Derivatives of Volumes and Pressures
C...
     3      /S/ DUMMY                   !   Spatial Derivatives if needed
     4      /R/ PI, PO, VO, TO, RUNIV,  ! Pi = 3.1412..., Std P,V,T& R
     *          PA2MM,
     *          KG, KP1, KOKM1, KP1OK, KOKP1,! K, K/(K-1), (K+1)/K, 1/K, 2/K, etc
     *          TWOOK, TMKOK, TWOOKP1,  ! Adiabatic constant and
     *          TWOOKM1, ONEOK,         ! derived constants
     *          CD,                     ! Orifice Discharge Coefficient
     *          TBODY, TAMB, PAMB,      ! Temperatures - Body & Ambient
     *          GMW, GMWS, GASK, GASKS, ! Gas parameters MW & Specif Heat ratios
     *          QAM, OMEGA, TINSP, FRM,  ! Breathing flow parameters
     *          TR, TPAUSE, RR, VTIDAL, ! Breathing flow parameters
     *          VFRCO, FRC, PPWBODY,    ! Initial, Current FRC, ppH2O Body
     *          AM(2), ANI(3), ANE(3),  ! Airway parameters
     *          VAO, VMAO, CINERT, DPAW,! Airway parameters
     *          AWI, DVL,               ! Airway parameters
     *          PREGO, PREF, Gdot1,     ! Regulator Outlet Pressure
     *          VDSMSK, AMIV, AMEV,     ! Mask parameters
     *          QIV, QEV,               ! Mask valve flow indicators
     *          VOL1, VOL2, VOL3, VOL4, ! Volumes
     *          AWN, AWO, AWH, AWAr,    ! Atomic wts
     *          GMN2, GMO2, GMCO2,      ! Molecular wts
     *          GMH2O, GMWAIR,          !    "     "
     *          FINSP(4), FEXP(4)       !Gas Fractions, O2,N2+Ar,CO2,H2O
     5      /I/ IP , IRELF            ! Print Counter
     6      /N/ NEQN
C...
C...  THE NUMBER OF DIFFERENTIAL EQUATIONS IS IN COMMON/N/ FOR USE IN
C...  SUBROUTINE FCN
C...
C...  COMPUTE DERIVATIVES
C...
C...   Define some statement Functions
```

Program MASKSUBS.FOR

```
C...
C...  Inspiratory and expiratory resistance of the mouth
C...
      DPMIE(Q) = 2.35D5*Q + 2.94D7*Q*Q ! Mouth dP vs Q for inspiration and expiration
C...                                   ! Volume Flow
C...  Inspiratory nose resistance
C...
      DPNI(Q)  = 2.94D5*Q + 2.94D8*Q*Q  ! Nose  dP vs Q for inspiration
C...                                    ! Volume flow
C...  Expiratory nose resistance
C...
      DPNE(Q)  = 2.94D5*Q + 3.92D8*Q*Q ! Nose  dP vs Q for expiration
C...                                   ! volume flow
      VTOT = VOL2 + VOL(3)  ! Lung Volume and Deadspace Volume
      TRD = TR + TPAUSE + TPAUSE/2.D0
      TC = TR + TPAUSE       ! Time per respiratory cycle
      RDF = 0.D0  ! RD Flag OFF
      IF(T .GE. TRD .AND. PAMB .GT. 6.68608D4) THEN   !RAPID DECOMPRESSION
         PAMB = DMAX1(1.01325D5
     *    - 2.D0*3.44643D4*(T - TRD)/TC, 6.68607D4)
         RDF = 1.D0  ! Set RD Flag ON
      END IF
      PREF = PAMB
      IF (PRESS(1) - PREF .GT. 5.D2 .AND. FLOW(1) .EQ. 0.D0) THEN
         PRESS(1)   = DMIN1(PREF + 5.D2, PRESS(1))! Simulate a 2 InWg relief valve
         PREGO = PRESS(1)
        ELSE
         PREGO = DMIN1(PREGO, PREF)
         PRESS(1) = DMIN1(PREGO,PRESS(1))
      END IF
      RHOAO = PRESS(2)*GMWAO/RUNIV/TBODY    ! Density Oronasal cavity
      IF (DMOD(T,TC) - TPAUSE .GE. 0.D0) THEN
           XT = DMOD(T,TC) - TPAUSE     ! Compute time from onset of
         ELSE                           ! Inspiration
           XT = 0.D0                    ! Pause time
      END IF
      IF(T .GT. TRD .AND. T .LT. 2.D0*TC + TPAUSE/2.D0) XT = 0.D0
      QI = 0.D0   ! Inspiratory Flag
      IF(XT .GT. 0.D0 .AND. XT .LE. TINSP) QI = 1.D0  ! Inspiratory Flag
      QE = 1.D0 - QI                                  ! Expiratory Flag
      i = 1
      DO WHILE (i .LE. NDE)
       PRESS(i) = DMAX1(PRESS(i), 1.D2)! Absolute pressure .GE. 100 Pa
         i = i + 1       ! Don't allow negative absolute pressure
      END DO
      IF(PRESS(2) .GT. 0.D0) FINSP(4) = PPWBODY/PRESS(2)   ! Mole Fraction H2O
      FINSP(2) = 1.D0 - (FINSP(1) + FINSP(3) + FINSP(4))   ! Inert Gas Fraction
      GMWAO = CALMW(FINSP)                   ! MW Oronasal Cavity Inspiration
      RHOAO = PRESS(3)*GMWAO/RUNIV/TBODY     ! Density Oronasal cavity
      GMWMH = FINSP(1)*GMO2 + FINSP(2)*GMN2 ! MW Mask Hose
      RHOMH = PRESS(1)*GMWMH/RUNIV/TAMB      ! Density Mask Hose [kg/m^3]
      OMF = 1.D0 - FRM ! Fraction Nose Breathing = 1 - Fraction Mouth Breathing
      IF( XT .GT. 0.D0 ) THEN
         DFDT(3) = QAM*OMEGA*DCOS(OMEGA*XT) ! Lung Volume Flow Derivative
         FLOW(3) = QAM*DSIN(OMEGA*XT)       ! Lung Volume Flow  [m^3/sec]
         Gdot3 = FLOW(3)*RHOAO              ! Lung Mass Flow [kg/sec]
         DP = DABS(FRM*DPMIE(FLOW(3)) + OMF*DPNI(FLOW(3))*QI ! Pressure
     *              + OMF*DPNE(FLOW(3))*QE)     ! loss through nose & mouth
         PRESS(3) = PRESS(2) - DP*QI + DP*QE     ! lung pressure
         C3 = VTOT*GMWAO/RUNIV/TBODY  ! Lung Capacitance
         IF(Gdot3 .NE. 0.D0) THEN
           R3 = DP/Gdot3    ! Airway Resistance
          ELSE
           R3 = 0.D0
         END IF
```

Program MASKSUBS.FOR

```
          DPDT(3) =  -(1.D0/C3*Gdot3 + DFDT(3)*R3) !  dP/dt for Lung
        ELSE
          FLOW(3) = 0.D0
          Gdot3 = 0.D0
          DVDT(3) = 0.D0
          DPDT(3) = 0.D0
          IF(FLOW(2) .EQ. 0.D0) PRESS(3) = PRESS(2)
          RHOAO = PRESS(2)*GMWAO/RUNIV/TBODY     ! Density Oronasal cavity
      END IF
      DPLUNG = PRESS(3) - PAMB   ! Differential Lung Pressure
C
C  Calculate the area of the inspiratory and expiratory valves
C  from curve fits to RAF P/Q Mask valve data (ASCC compliant)
C
C          !  Exp valve has 0.5 InWg cracking Pressure
       IF(PRESS(2) .GT. (PRESS(1) + 50.D0) ) THEN
          QIV = 0.D0
          PRESS(1) = DMAX1(PREGO, PRESS(1))
          RHOMH = PRESS(1)*GMWMH/RUNIV/TAMB  ! Density Mask Hose [kg/m^3]
          FLOW(1) = 0.D0
          Gdot1 = 0.D0
          DPDT(1) = 0.D0
          FEXP(2) = 1.D0 - (FEXP(1) + FEXP(3) + FEXP(4))  ! Inert gas fraction
          GMWE = CALMW(FEXP)   ! Expired gas molecular weight
          PEXPV = PAMB + DMAX1(0.D0, PRESS(1) - PAMB) + 50.D0 ! 0.5 Inch Wg Spring Pressure
               ! Compensation pressure
               ! Expiratory valve back pressure can be tailored to
               ! compensation characteristics of a particular valve
          DPX  = DMAX1(PRESS(2) - PEXPV,0.D0)  ! Delta-P Exp Valve
          AMEV = DMAX1(0.D0, DMIN1(3.84D-7*DPX -1.11D-5,1.6D-4))!Area Expiratory Valve
C                            PMASK     PMHOSE
          CALL ORIFLO(CD, AMEV, PRESS(2), PEXPV , TBODY, GMWE,
     *                     FLOW(2), Gdot2, PCRIT)    ! Expiratory valve flow
          FLOW(2) = DMIN1(-FLOW(2), 0.D0)  ! Flow is reversed for expiration
          IF(FLOW(2) .LT. 0.D0) THEN
             QEV = 1.0
            ELSE
             QEV = 0.D0
          END IF
          DPDT(2) = PRESS(2)/VTOT*(FLOW(2) - FLOW(3))  ! Mask dP/dt.
        ELSEIF(PRESS(1) .GT. (PRESS(2) + 10.D0)) THEN
C           IF (FLOW(1) .NE. 0.D0) THEN
C              PRESS(1) = DMIN1(PRESS(1),PREF)
C             ELSE
C              PRESS(1) = DMIN1(PRESS(1), PREGO)
C           END IF
          DPI  = PRESS(1) - PRESS(2) !Delta-P Inspiratory Valve
          AMIV = DMAX1(0.D0, DMIN1(1.009D-6*DPI,1.5D-4)) !Area Insp Valve
          GMWI = CALMW(FINSP) ! Inspiratory gas molecular weight
C                              PMHOSE    PMASK
          CALL ORIFLO(CD, AMIV, PRESS(1), PRESS(2), TAMB, GMWI,
     *                     FLOW(2), Gdot2, PCRIT)    ! Inspiratory valve flow
          FLOW(2) = DMAX1(FLOW(2), 0.D0)
          IF(FLOW(2) .GT. 0.D0) THEN
             QIV = 1.0
            ELSE
             QIV = 0.D0
          END IF
          RHOAO = PRESS(2)*GMWAO/RUNIV/TBODY     ! Density Oronasal cavity
          RHOMH = PRESS(1)*GMWMH/RUNIV/TAMB      ! Density Mask Hose [kg/m^3]
          Gdot1 =  FLOW(2)*RHOAO ! Mass flow into mask
          FLOW(1) = Gdot1/RHOMH  ! Volume flow from mask tube
          DPH = DPHOSE(Gdot1)    ! Pressure loss in hose
          PREGO = DMIN1(PRESS(1) + DPH, PREF) ! Regulator outlet pressure
          DELTAP = PREGO - PREF  ! Differential pressure relative to reference
```

Program MASKSUBS.FOR

```
            DPDT(1) = 1.D0/VOL1*(PREGO*GTREG(DELTAP)/RHOMH    ! GTREG is mass flow v DP
     *               - PRESS(1)*FLOW(1)) ! dP/dt for mask hose outlet pressure
            DPDT(2) = PRESS(2)/VTOT*(FLOW(2) - FLOW(3)) ! dP/dt for Mask
          ELSE
            IF( QIV .EQ. 0.D0 ) THEN
               FLOW(1) = 0.D0
               Gdot1   = 0.D0
               DPDT(1) = 0.D0
               DFDT(1) = 0.D0
               PREGO = PREF
               PRESS(1) = DMIN1(PREGO, PRESS(1))
               RHOMH = PRESS(1)*GMWMH/RUNIV/TAMB      ! Density Mask Hose [kg/m^3]
            END IF
            IF(QIV + QEV .EQ. 0.D0) THEN
              FLOW(2) = 0.D0
              DFDT(2) = 0.D0
              IF(FLOW(3) .EQ. 0.D0) PRESS(3) = PRESS(2)
              DPDT(2) = -PRESS(2)/VTOT*FLOW(3) ! dP/dt for Mask
            END IF
        END IF
        DVDT(1) = FLOW(1)  ! Cumulative volume flow from regulator
        DVDT(2) = FLOW(2)  ! Flow is derivative of volume
        DVDT(3) = FLOW(3)  ! Flow is derivative of volume
 1000   RETURN
        END
        SUBROUTINE PRINT(NI,NO)
C...
C...    Prints selected output variables at the specified time interval.
C...
C...
C...    ODE COMMON
C...
C...     /Y/ time variables
C...     /F/ time derivatives of variables
C...     /S/ spatial derivatives of variables
C...     /R/ & /I/ real and integer parameters required to define constants and
C...              define the spatial integration grid.
C...
        IMPLICIT DOUBLE PRECISION (A-H, O-Z)
        PARAMETER (NDE=3)
        DOUBLE PRECISION  KG, KP1, KP1OK, KOKM1, KOKP1
        INTEGER NSTOP, NORUN,  IP, NEQN
        COMMON/T/         T, XT,     NSTOP,     NORUN
       1      /Y/ VOL(NDE), FLOW(NDE), PRESS(NDE),       !
C...
C...            VOL(1)  [=]  Cumulative volume flow from regulator
C...            VOL(2)  [=]  Cumulative volume flow into mask
C...            VOL(3)  [=]  Cumulative Respired volume (Insp - Exp)
C...
C...            FLOW(1) [=]  Instantaneous flow from regulator [m^3/sec]
C...            FLOW(2) [=]  Instantaneous mask flow       [m^3/sec]
C...            FLOW(3) [=]  Instantaneous Oro-nasal flow  [m^3/sec]
C...
C...            PRESS(1) [=]  Regulator Outlet Pressure
C...            PRESS(2) [=]  Mask hose pressure
C...            PRESS(3) [=]  Mask cavity Pressure = Intraoral pressure
C...
       2      /F/ DVDT(NDE), DFDT(NDE), DPDT(NDE),
C...
C...        Derivatives of Volumes and Pressures
C...
       3      /S/ DUMMY                     !   Spatial Derivatives if needed
       4      /R/ PI, PO, VO, TO, RUNIV,  ! Pi = 3.1412..., Std P,V,T& R
       *          PAZMM,
       *          KG, KP1, KOKM1, KP1OK, KOKP1,! K, K/(K-1), (K+1)/K, 1/K, 2/K, etc
```

```
     *          TWOOK, TMKOK, TWOOKP1,  ! Adiabatic constant and
     *          TWOOKM1, ONEOK,         ! derived constants
     *          CD,                     ! Orifice Discharge Coefficient
     *          TBODY, TAMB, PAMB,      ! Temperatures - Body & Ambient
     *          GMW, GMWS, GASK, GASKS, ! Gas parameters MW & Specif Heat ratios
     *          GAM, OMEGA, TINSP, FRM, ! Breathing flow parameters
     *          TR, TPAUSE, RR, VTIDAL, ! Breathing flow parameters
     *          VFRCO, FRC, PPWBODY,   ! Initial, Current FRC, ppH2O Body
     *          AM(2), ANI(3), ANE(3),  ! Airway parameters
     *          VAO, VMAO, CINERT, DPAW,! Airway parameters
     *          AWI, DVL,               ! Airway parameters
     *          PREGO, PREF, Gdot1,     ! Regulator Outlet Pressure
     *          VDSMSK, AMIV, AMEV,     ! Mask parameters
     *          QIV, QEV,               ! Mask valve flow indicators
     *          VOL1, VOL2, VOL3, VOL4, ! Volumes
     *          AWN, AWO, AWH, AWAr,    ! Atomic wts
     *          GMN2, GMO2, GMCO2,      ! Molecular wts
     *          GMH2O, GMWAIR,           !    "      "
     *          FINSP(4), FEXP(4)       !Gas Fractions, O2,N2+Ar,CO2,H2O
     5     /I/ IP , IRELF             ! Print Counter
     6     /N/ NEON
C...
C...
C...  PRINT A HEADING FOR THE NUMERICAL SOLUTION
C...
        IP = IP + 1
C       WRITE(NO,2)
C       WRITE(NO,2) T
C...
C...  PRINT THE SOLUTION
C...
       WRITE( *,'(1X,2F10.3)') T, XT
       WRITE( *,3) T,  PREGO, (PRESS(k), k=1,NDE), (DPDT(k), k=1,NDE)
       WRITE( *,3) T,  (VOL(k), k=1,NDE),   (FLOW(k), k=1,NDE)
       WRITE(NO,2) T,  PAMB, PREGO, (PRESS(k), k=1,NDE),
     *   (DPDT(k), k=1,NDE), (VOL(k), k=1,NDE),   (FLOW(k), k=1,NDE)
 2    FORMAT(F8.4, 16E17.8)
 3    FORMAT(1X,F8.4, 30E13.5)
      RETURN
      END
      DOUBLE PRECISION FUNCTION GTREG(DP)
      DOUBLE PRECISION DP
C...
C...  REGULATOR MASS FLOW VS OUTLET PRESSURE
C...  BASED ON CURVE FIT
C...
      IF(DP .GE. 0.D0) THEN
         GTREG = 0.D0
         RETURN
      END IF
      GTREG = DMAX1(1.D-10,-1.78D-3 - 4.985D-5*DP - 1.585D-7*DP**2
     *             - 2.136D-10*DP**3)
      IF(DP .LT. -3.5D2) GTREG = 5.D-3
      RETURN
      END
      DOUBLE PRECISION FUNCTION POREG(Gt)
      DOUBLE PRECISION  Gt
C...
C...  Regulator differential outlet pressure = f(mass flow at outlet)
C...  Units:  Pa vs kg/sec
C...
      POREG = -46.72D0 - 1.219D4*Gt - 1.009D7*Gt*Gt  ! Pascals
      IF(Gt .LT. 0.D0) POREG = 0.D0
      RETURN
      END
```

Program MASKSUBS.FOR

```
      DOUBLE PRECISION FUNCTION DPHOSE(Gt)
C
C...  Mask hose Resistance
C...
      DOUBLE PRECISION Gt
      DPHOSE =      1.1462D4*Gt + 5.25D6*Gt*Gt ! delta-P hose vs volume flow
      RETURN
      END
      SUBROUTINE ORIFLO(CDORIF, AREA, PUP, PDWN, TABS, GMWT,
     *  FLOWVOL, FLOWMASS, PCRIT)
C
C**  Subroutine ORIFLO uses the compressible flow equations for an ideal
C     gas to compute the flow through an orifice given the upstream and
C     downstream conditions, the orifice area, and the gas parameters.
C
C
C    PARAMETERS          DESCRIPTION                           UNITS
C
C       CD       [=]  The orifice discharge coefficient         [UNITLESS]
C       AREA     [=]  The area of the orifice                   [SQ METERS]
C       TABS     [=]  The absolute temperature                  [DEG KELVIN]
C       GMW      [=]  The molecular weight of the gas           [KG/KG-MOL]
C       GASK     [=]  The adiabatic exponent ratio of specific
C                     heats of the gas (Cp/Cv)                  [UNITLESS]
C       RUNIV    [=]  Universal Gas Constant                    [NEWTON*METERS/KGMOL/DEG K]
C
c    VARIABLES
C
C       PUP      [=]  Upstream absolute pressure                [PASCALS]
C       PDwn     [=]  Downstream absolute pressure              [PASCALS]
C
C******OUTPUT PARAMETERS
C
C       FLOWVOL  [=] Volumetric flow                            [CU METERS/SEC]
C       FLOWMASS [=] Mass flow                                  [KG/SEC]
C       PCRIT    [=] Critical downstream pressure for sonic flow.[PASCALS]
C
C*********NOTE:  If other units are desired, the universal gas constant
C
C       RUNIV must be changed to the correct value for the chosen units.
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION  KG, KP1, KP1OK, KOKM1, KOKP1
      COMMON /R/ PI, PO, VO, TO, RUNIV, ! Pi = 3.1412..., Std P,V,T& R
     *          PA2MM,
     *          KG, KP1, KOKM1, KP1OK, KOKP1,! K, K/(K-1), (K+1)/K, 1/K, 2/K, etc
     *          TWOOK, TMKOK, TWOOKP1,  ! Adiabatic constant and
     *          TWOOKM1, ONEOK,         ! derived constants
     *          CD,                     ! Orifice Discharge Coefficient
     *          TBODY, TAMB, PAMB,            ! Temperatures - Body & Ambient
     *          GMW, GMWS, GASK, GASKS, ! Gas parameters MW & Specif Heat ratios
     *          QAM, OMEGA, TINSP, FRM,      ! Breathing flow parameters
     *          TR, TPAUSE, RR, VTIDAL, ! Breathing flow parameters
     *          VFRCO, FRC, PPWBODY,    ! Initial, Current FRC, ppH2O Body
     *          AM(2), ANI(3), ANE(3),  ! Airway parameters
     *          VAO, VMAO, CINERT, DPAW,! Airway parameters
     *          AWI, DVL,
     *          PREGO, PREF, Gdot1,     ! Regulator Outlet Pressure
     *          VDSMSK, AMIV, AMEV,     ! Mask parameters
     *          QIV, QEV,               ! Mask valve flow indicators
     *          VOL1, VOL2, VOL3, VOL4, ! Volumes
     *          AWN, AWO, AWH, AWAr,    ! Atomic wts
     *          GMN2, GMO2, GMCO2,      ! Molecular wts
     *          GMH2O, GMWAIR,          !    "     "
     *          FINSP(4), FEXP(4)       !Gas Fractions, O2,N2+Ar,CO2,H2O
```

Program MASKSUBS.FOR

```
      COMMON /I/ IP , IRELF
C
C  PRESERVE THE CALLING CONSTANTS AND VARIABLES.
C
C
C  THESE CONSTANTS DO NOT CHANGE DURING A SINGLE CALL,
C  BUT MAY CHANGE BETWEEN CALLS.
C
      IF(PUP .EQ. PDWN .OR. PUP .LE. 0.D0 .OR. TABS .LE. 0.D0 .OR.
     *   PDWN .LE. 0.D0 .OR. AREA .LE. 0.D0 .OR. GMWT .LE. 0.D0) THEN
        FLOWMASS = 0.0D0
        FLOWVOL  = 0.0D0
        RETURN
      END IF
      WT = GMWT
      PU = PUP
      PD = PDWN
      PR = PD/PU
      C  = CDORIF
      A  = AREA
      T  = TABS
      RGAS = RUNIV/WT               ! SPECIFIC GAS CONSTANT
      FWD = 1.D0                    ! FORWARD FLOW FLAG
      IF (PR .GT. 1.0D0) THEN
         FWD = -FWD                 ! REVERSE FLOW
         PR = 1.0D0/PR              ! EXCHANGE PU & PD
         PTEMP = PU
         PU = PD
         PD = PTEMP
      END IF
      RHOU = PU/(RGAS*T)            ! UPSTREAM GAS DENSITY
      PCRIT = TWOOKP1**KOKM1*PU     ! CRITICAL PRESSURE FOR SONIC FLOW
C
C** IF DOWNSTREAM CONDITIONS < PCRIT USE SONIC EQUATIONS
C
   99 FORMAT(5X,5E20.8)
      IF(PD .LT. PCRIT) THEN
C
C     ** SONIC CONDITIONS APPLY IF PD < .OR. = PCRIT
C
         FACT = 2.0D0*KOKP1
         ARG = PU*RHOU*FACT*TWOOKP1**TWOOKM1
         IF(ARG .LT. 0.D0) PAUSE 'INVALID ARGUMENT IN ORIFLOW'
         FLOWMASS = FWD*C*A*DSQRT(ARG)
         FLOWVOL  = FLOWMASS/RHOU
       ELSE
C
C     ** ELSE USE THE SUBSONIC EQUATIONS
C
         FACT = 2.0D0*KOKM1
         ST = DSQRT(T)
         ARG =  FACT/RGAS*(PR**TWOOK - PR**KP1OK)
         IF(ARG .LT. 0.D0) PAUSE 'INVALID ARGUMENT IN ORIFLOW'
         FLOWMASS = FWD*C*A/ST*PU*DSQRT(ARG)
         FLOWVOL = FLOWMASS/RHOU
      END IF
      RETURN
      END
      SUBROUTINE DICAN(IATOP,ALT,PABS,TEMPK)
C
C  MODIFIED 7/12/87 TO INCLUDE IMPROVEMENTS PROGRAMMED BY MR L. GILL OF
C  MOOG CARLETON GROUP.
C
C  DOUBLE PRECISION VERSION CREATED 12/1/93 FOR MASKMODEL
C
```

Program MASKSUBS.FOR

```
C  ICAN COMPUTES THE ABSOLUTE PRESSURE FOR A GIVEN ALTITUDE
C    (OR VICE VERSA) BY ICAN STANDARD ATMOSPHERE MODEL(1953).
C
C       TO COMPUTE PRESSURE FROM ALTITUDE SET IATOP .GE. 0
C       TO COMPUTE ALTITUDE FROM PRESSURE SET IATOP .LT. 0
C
C    UNITS ON ALTITUDE RETURNED IN FEET.
C
C    THE ABSOLUTE TEMPERATURE ESTIMATE FOR THE ALTITUDE IS
C    RETURNED IN DEG KELVIN(TEMPK).
C
C    THE MODEL IS MOST ACCURATE BETWEEN -1,000 AND 100,000 FEET.
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      INTEGER IATOP
      X = 5.256D0
      TC = 6.5D-3
      TO = 288.16D0
      SCALE = 1.0D-5
      IF(IATOP .LT. 0) GO TO 200
      ALT = ALT/1000.D0
C
C  CHECK FOR TROPOPAUSE
C
      IF(ALT .GT. 36.D0) GO TO 150
C
C  MODEL FOR BELOW TROPOPAUSE
C
      TEMPK = TO - TC*ALT
      PABS = ((TO-1.9812D0*ALT)/TO)**X+(2.15287D0*ALT-8.16554D0)*SCALE
      GO TO 999
C
C  MODEL FOR ABOVE TROPOPAUSE
C
  150 C = 0.224D0*10.D0**((36.0889D0 - ALT)/47.89968D0)
      IF(ALT .LT. 50.D0) B = (2.52D0*ALT - 90.5D0)*SCALE
      IF(ALT .GE. 50.D0) B = (0.2284D0*ALT + 21.775D0)*SCALE
      PABS = C + B
      TEMPK = 273.2D0 - 56.5D0
      GO TO 999
C
C  COMPUTE ALTITUDE FROM PRESSURE
C  CHECK FOR TROPOPAUSE
C
  200 IF(PABS .GE. .001D0) GOTO 205
      ALT = 170.D0
      GOTO 999
  205 IF(PABS .LT. 0.2236D0) GO TO 250
C
C   MODEL FOR BELOW TROPOPAUSE
C
      ALTOLD = TO/TC*(1.D0 - PABS**(1/X))/1000.D0*3.281D0
  210 ARG = (2.15287D0*ALTOLD - 8.16554D0)*SCALE
      ALT = TO/1.9812D0*(1.D0 - (PABS - ARG)**(1.D0/X))
      TEMPK = TO - TC*ALT
      IF(ABS(ALT - ALTOLD) .LT. 1.D-4) GO TO 999
      ALTOLD = (ALT + ALTOLD)/2.D0
      GO TO 210
C
C   MODEL FOR ABOVE TROPOPAUSE
C
C     CHECK FOR 50000 FEET OR ABOVE.
C
  250 ALTOLD = (DLOG10(0.2236D0/PABS)*14.6D0 + 11.D0)*3.281D0
  260 B = (2.52D0*ALTOLD - 90.5D0)*SCALE
```

Program MASKSUBS.FOR

```
      IF(PABS .LT. 0.11511600) B = (0.228400*ALTOLD+21.77500)*SCALE
      ALT = 36.088900 - 47.8996800*DLOG10((PABS - B)/0.22400)
      TEMPK = 273.200 - 56.500
      IF(ABS(ALT - ALTOLD) .LT. 1.D-4) GO TO 999
      ALTOLD = (ALT + ALTOLD)/2.D0
      GO TO 260
  999 ALT = ALT*1000.D0
      RETURN
      END
      SUBROUTINE MILCAB(PABACF,PABCAB,PMAX,IUNITS)
C
C  DOUBLE PRECISION VERSION
C
C
C       U.S MILSPEC CABIN PRESSURIZATION SCHEDULE AS DETAILED
C       IN MIL-E-38453A (USAF) 2 DEC 1971.
C
C    GIVEN ABSOLUTE PRESSURE OF AIRCRAFT MILCAB RETURNS
C    THE CABIN ABSOLUTE PRESSURE OR VICE VERSA.
C      TO COMPUTE CABIN FROM ACFT SET INVERT .GE. 0.
C
C   PRESSURE UNITS SET BY IUNITS
C     SET IUNITS = 0 FOR PSIA
C     SET IUNITS = 1 FOR MM HG
C
C   PMAX = THE MAXIMUM DIFFERENTIAL PRESSURE PRODUCED BY THE ECS
C
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      INTEGER IUNITS
      CPMM = 51.72D0
      IF(IUNITS .EQ. 1) PABACF = PABACF/CPMM
      IF(PABACF .GT. 10.92D0) GO TO 110
      GO TO 120
  110 PABCAB = PABACF
      GO TO 999
  120 IF(PABACF .LE. 10.917D0 - PMAX) GO TO 130
      PABCAB = 10.917D0
      GO TO 999
  130 PABCAB = PABACF + PMAX
  999 IF(IUNITS .NE. 1) RETURN
      PABCAB = PABCAB*CPMM
      PABACF = PABACF*CPMM
      RETURN
      END
      DOUBLE PRECISION FUNCTION PBFORA(PCH)
C
C  COMPUTES PRESSURE BREATHING FOR ALTITUDE AS A FUNCTION OF AMBIENT PRESS
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IF(PCH .GT. 0.2D0) THEN
      PBFORA = 0.0D0
      ELSE
      PBFORA = DMAX1(0.04D0,0.094D0-0.49D0*PCH)
      END IF
      RETURN
      END
      DOUBLE PRECISION FUNCTION DVL(DPLUNG)
C
C  COMPUTES FUNCTIONAL RESIDUAL VOLUME AS FUNCTION
C  OF DIFFERENTIAL BREATHING PRESSURE
C
      DOUBLE PRECISION DPLUNG
      DVL =  2.2502D-7*DPLUNG
```

```
      RETURN
      END
      SUBROUTINE PULFCN(TIME,PPBALT,VDOTE,QPEAK,QRESP,VLUNG)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DOUBLE PRECISION  KG, KP1, KP1OK, KOKM1, KOKP1
      COMMON/T/         T, XT,    NSTOP,     NORUN
      COMMON /R/ PI, PO, VO, TO, RUNIV,  ! Pi = 3.1412..., Std P,V,T& R
     *         PA2MM,
     *         KG, KP1, KOKM1, KP1OK, KOKP1,! K, K/(K-1), (K+1)/K, 1/K, 2/K, etc
     *         TWOOK, THKOK, TWOOKP1,  ! Adiabatic constant and
     *         TWOOKM1, ONEOK,         ! derived constants
     *         CD,                     ! Orifice Discharge Coefficient
     *         TBODY, TAMB, PAMB,           ! Temperatures - Body & Ambient
     *         GMW, GMWS, GASK, GASKS, ! Gas parameters MW & Specif Heat ratios
     *         QAM, OMEGA, TINSP, FRM,      ! Breathing flow parameters
     *         TR, TPAUSE, RR, VTIDAL, ! Breathing flow parameters
     *         VFRCO, FRC, PPWBODY,    ! Initial, Current FRC, ppH2O Body
     *         AM(2), AMI(3), AME(3),  ! Airway parameters
     *         VAO, VMAO, CINERT, DPAW,! Airway parameters
     *         AWI, DVL,
     *         PREGO, PREF, Gdot1,     ! Regulator Outlet Pressure
     *         VDSMSK, AMIV, AMEV,     ! Mask parameters
     *         QIV, QEV,               ! Mask valve flow indicators
     *         VOL1, VOL2, VOL3, VOL4, ! Volumes
     *         AWN, AWO, AWH, AWAr,    ! Atomic wts
     *         GMN2, GMO2, GMCO2,      ! Molecular wts
     *         GMH2O, GMAIR,           !    "     "
     *         FINSP(4), FEXP(4)       !Gas Fractions, O2,N2+Ar,CO2,H2O
C
      QOFT(T,QM,W) = QM*DSIN(W*T)
      VLOFT(T,QM,W) = QM/W*(1.0D0-DCOS(W*T))
      VFRC(PB) = 2.5D0 + 0.1D0*PB*760.D0
      TWOPI = 2.D0*PI
C
C   PEAK FLOW = VDOTE*PI FOR A HALF WAVE RECTIFIED SINUSOIDAL
C   DEMAND FLOW PATTERN.
C
      QPEAK = VDOTE*PI
C
C   USE THE CURVE FIT VTID TO ESTIMATE TIDAL VOLUME FROM AVERAGE FLOW.
C   THEN DIVIDE BY VE TO GET TAV, THE AVERAGE TIME PER BREATHING CYCLE.
C
      CALL VTIDL(VDOTE,VTIDAL)
      TAV = VTIDAL/VDOTE
C
C   COMPUTE RESPIRATION RATE FROM  RESP RATE (BREATHS/TIME) = 1/TAV
C
      RR = 1.D0/TAV
C
C   SCALE TIME FROM MINUTES TO SECONDS/CYCLE
C
      TAV = TAV*60.D0
C
C   COMPUTE THE AVERAGE RADIAN BREATHING FREQUENCY
C
      OMEGA = TWOPI/TAV
      QRESP = QOFT(TIME,QPEAK,OMEGA)
      VLUNG = VFRC(PPBALT) + VLOFT(TIME,QPEAK,OMEGA)/60.D0
      RETURN
      END
C...
C...
C...
      SUBROUTINE VTIDL(VE,VT)
C   DOUBLE PRECISION VERSION
```

Program MASKSUBS.FOR

```
C
C  ESTIMATE TIDAL VOLUME FROM PULMONARY VENTILATION (V DOT E) ACCORDING TO
C  A THE MODEL OF HEY ET AL, RESPIRATION PHYSIOLOGY (1966) V1,191-205.
C
C
C   USE HEY'S MEAN MODEL PARAMETERS FOR "M" AND "K"
C   (MBAR = 28 +/- 2 [1/MIN] AND KBAR = 0.31 +/- 0.08 [LITERS])
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DOUBLE PRECISION MBAR,KBAR
      MBAR = 28.D0
      KBAR = 0.31D0
      VT = VE/MBAR + KBAR
      RETURN
      END
C...
C...
C...
      SUBROUTINE FIOX(PAO2,PACO2,R,PAB,FIO2,FIO2MX,PINSP,PAOXY)
C
C   DOUBLE PRECISION VERSION
C
C   FIOX EMPLOYS THE ALVEOLAR AIR EQUATION FOR STEADY STATE EXCHANGE
C   TO CALCULATE THE MINIMUM FRACTION INSPIRED O2 (FIO2) REQUIRED
C   TO MAINTAIN PAO2 GIVEN PACO2, R, AND ABSOLUTE PRESSURE(PAB).
C   PRESSURES MUST BE IN MM HG. IF PRESSURE BREATHING IS REQUIRED
C   FIO2 IS SET TO 1.0 AND THE REQUIRED SAFETY PRESSURE IS RETURNED IN PINSP
C   IN MM HG(GAUGE).  THE ESTIMATED ALVEOLAR OXYGEN PARTIAL PRESSURE WITHOUT
C   SAFETY PRESSURE IS RETURNED IN PAOXY.
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      OMROR = (1.D0 - R)/R
      A = PAO2 + PACO2/R
      B = PACO2*OMROR + PAB - 47.D0
      FIO2 = A/B
      PAOXY = PAO2
      PINSP = 0.0D0
      IF(FIO2 .GE. 0.D0 .AND. FIO2 .LE. FIO2MX) RETURN
      IF(FIO2 .LT. 0.D0) GO TO 999
      PC = 47.D0 - PACO2*OMROR + A/FIO2MX
      PINSP = PC - PAB
      FIO2 = FIO2MX
      BNEW = PACO2*OMROR + PC - 47.D0
      PAOXY = FIO2MX*BNEW - PACO2/R
      RETURN
  999 WRITE(*,2) FIO2
      FIO2 = 0.
    2 FORMAT(5X,'***ERROR IN FIOX, FIO2 = ',1PE12.3)
      RETURN
      END
C...
C...
C...
      SUBROUTINE RDO2(PAO2MN,FRIO2,FIO2MX,PABACF,PABCAB,PSAFTI,
     *PSAFTF,PACO2,R)
C
C   DOUBLE PRECISION VERSION
C
C   RDO2 COMPUTES MINIMUM FRACTION INSP O2 REQUIRED TO MAINTAIN
C   30MM HG PAO2 FOLLOWING DECOMPRESSION FROM PABCAB TO PABACF.
C   CORRECTION FOR SAFETY PRESSURE IS MADE. REQUIRES
C   SUBROUTINE FIO2(PAO2,PACO2,R,PABCAB,FIO2,FIO2MX,PINSP,PAOXY)
C   PRESSURES MUST BE SUPPLIED IN MM HG.  PABACF, PABCAB & PACO2
C   ARE ABSOLUTE PRESSURES, PSAFTI IS GAUGE PRESSURE(MASK LESS CABIN).
C   BEFORE DECOMPRESSION PSAFTF IS SAFETY PRESSURE AFTER DECOMPRESSION.
```

Program MASKSUBS.FOR

```
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      PLI = PABCAB + PSAFTI - 47.D0   ! mmHg
      PLF = PABACF + PSAFTF - 47.D0   ! mmHg
      PAO2 = PAO2MN*PLI/PLF
      CALL FIOX(PAO2,PACO2,R,PABCAB+PSAFTI,FRIO2,FIO2MX,PINSP,PAOXY)
      RETURN
      END
C...
C...
C...
      SUBROUTINE ALVPO2(FIO2,PACO2,R,PAB,PINSP,PAOXY,EQVALT)
C
C     DOUBLE PRECISION VERSION
C
C    SUBROUTINE ALVPO2 EMPLOYS THE ALVEOLAR AIR EQUATION TO ESTIMATE
C    THE STEADY STATE ALVEOLAR OXYGEN PARTIAL PRESSURE (PAOXY), THE
C    CORRESPONDING ALVEOLAR CARBON DIOXIDE PARTIAL PRESSURE (PACO2)
C    AND RESPIRATORY EXCHANGE RATIO, (R) GIVEN THE FRACTION INSPIRED
C    OXYGEN (FIO2), THE ABSOLUTE AMBIENT PRESSURE (PAB) AND THE LEVEL OF
C    DIFFERENTIAL PRESSURE BREATHING (PINSP).  THE SUBROUTINE ALSO
C    CALCULATES THE ALTITUDE (EQVALT) AT WHICH THE EQUIVALENT PAOXY IS
C    PRODUCED IF AIR IS BREATHED.
C
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
  100 SAVPO2 = PAOXY
      SAVCO2 = PACO2
      OMROR = (1.0D0 - R)/R
      PABRSP = PAB + PINSP
      PAOXY = FIO2*(PACO2*OMROR + PABRSP - 47.D0) - PACO2/R
      IF(PAOXY .LE. 0.0D0) THEN
      PAOXY = FIO2*(PABRSP - 47) - PACO2
      ELSE
      PACO2 = 1.D0/(0.0207D0 + 0.474D0/PAOXY)
      END IF
      IF(PAOXY .LE. 25.D0) THEN
         R = 1.05D0
      ELSE
         R = 5.D0*DLOG(PACO2)-0.7843D0*ALOG(PACO2)**2.D0-6.9266D0
      END IF
  500 IF(ABS(SAVPO2 - PAOXY) .LE. 0.01D0) THEN
       GO TO 1000
      ELSE
       PAOXY = (SAVPO2+PAOXY)/2.D0
       PACO2 = (SAVCO2+PACO2)/2.D0
       GO TO 100
      END IF
 1000 PABRSP = (PAOXY + PACO2/R)/0.2095D0 - PACO2*OMROR + 47.D0
      PABRSP = PABRSP/759.9D0
      CALL DICAN(-1,EQVALT,PABRSP,TEMPK)
      RETURN
      END
```

Program RKF45.FOR

```
*DECK RKF45
      SUBROUTINE RKF45(F,NEQN,Y,T,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
C
C     FEHLBERG FOURTH-FIFTH ORDER RUNGE-KUTTA METHOD
C
C     WRITTEN BY H.A.WATTS AND L.F.SHAMPINE
C                   SANDIA LABORATORIES
C                  ALBUQUERQUE,NEW MEXICO
C
C    RKF45 IS PRIMARILY DESIGNED TO SOLVE NON-STIFF AND MILDLY STIFF
C    DIFFERENTIAL EQUATIONS WHEN DERIVATIVE EVALUATIONS ARE INEXPENSIVE.
C    RKF45 SHOULD GENERALLY NOT BE USED WHEN THE USER IS DEMANDING
C    HIGH ACCURACY.
C
C ABSTRACT
C
C    SUBROUTINE  RKF45  INTEGRATES A SYSTEM OF NEQN FIRST ORDER
C    ORDINARY DIFFERENTIAL EQUATIONS OF THE FORM
C             DY(I)/DT = F(T,Y(1),Y(2),...,Y(NEQN))
C              WHERE THE Y(I) ARE GIVEN AT T .
C    TYPICALLY THE SUBROUTINE IS USED TO INTEGRATE FROM T TO TOUT BUT IT
C    CAN BE USED AS A ONE-STEP INTEGRATOR TO ADVANCE THE SOLUTION A
C    SINGLE STEP IN THE DIRECTION OF TOUT.  ON RETURN THE PARAMETERS IN
C    THE CALL LIST ARE SET FOR CONTINUING THE INTEGRATION. THE USER HAS
C    ONLY TO CALL RKF45 AGAIN (AND PERHAPS DEFINE A NEW VALUE FOR TOUT).
C    ACTUALLY, RKF45 IS AN INTERFACING ROUTINE WHICH CALLS SUBROUTINE
C    RKFS FOR THE SOLUTION.  RKFS IN TURN CALLS SUBROUTINE  FEHL WHICH
C    COMPUTES AN APPROXIMATE SOLUTION OVER ONE STEP.
C
C    RKF45  USES THE RUNGE-KUTTA-FEHLBERG (4,5)  METHOD DESCRIBED
C    IN THE REFERENCE
C    E.FEHLBERG , LOW-ORDER CLASSICAL RUNGE-KUTTA FORMULAS WITH STEPSIZE
C                 CONTROL , NASA TR R-315
C
C    THE PERFORMANCE OF RKF45 IS ILLUSTRATED IN THE REFERENCE
C    L.F.SHAMPINE,H.A.WATTS,S.DAVENPORT, SOLVING NON-STIFF ORDINARY
C                 DIFFERENTIAL EQUATIONS-THE STATE OF THE ART ,
C                 SANDIA LABORATORIES REPORT SAND75-0182 ,
C                 TO APPEAR IN SIAM REVIEW.
C
C
C    THE PARAMETERS REPRESENT-
C      F -- SUBROUTINE F(T,Y,YP) TO EVALUATE DERIVATIVES YP(I)=DY(I)/DT
C      NEQN -- NUMBER OF EQUATIONS TO BE INTEGRATED
C      Y(*) -- SOLUTION VECTOR AT T
C      T -- INDEPENDENT VARIABLE
C      TOUT -- OUTPUT POINT AT WHICH SOLUTION IS DESIRED
C      RELERR,ABSERR -- RELATIVE AND ABSOLUTE ERROR TOLERANCES FOR LOCAL
C            ERROR TEST. AT EACH STEP THE CODE REQUIRES THAT
C                 ABS(LOCAL ERROR) .LE. RELERR*ABS(Y) + ABSERR
C            FOR EACH COMPONENT OF THE LOCAL ERROR AND SOLUTION VECTORS
C      IFLAG -- INDICATOR FOR STATUS OF INTEGRATION
C      WORK(*) -- ARRAY TO HOLD INFORMATION INTERNAL TO RKF45 WHICH IS
C            NECESSARY FOR SUBSEQUENT CALLS. MUST BE DIMENSIONED
C            AT LEAST  3+6*NEQN
C      IWORK(*) -- INTEGER ARRAY USED TO HOLD INFORMATION INTERNAL TO
C            RKF45 WHICH IS NECESSARY FOR SUBSEQUENT CALLS. MUST BE
C            DIMENSIONED AT LEAST  5
C
C
C  FIRST CALL TO RKF45
C
C    THE USER MUST PROVIDE STORAGE IN HIS CALLING PROGRAM FOR THE ARRAYS
C    IN THE CALL LIST  -      Y(NEQN) , WORK(3+6*NEQN) , IWORK(5)  ,
C    DECLARE F IN AN EXTERNAL STATEMENT, SUPPLY SUBROUTINE F(T,Y,YP) AND
```

Program RKF45.FOR

```
C    INITIALIZE THE FOLLOWING PARAMETERS-
C
C      NEQN -- NUMBER OF EQUATIONS TO BE INTEGRATED.  (NEQN .GE. 1)
C      Y(*) -- VECTOR OF INITIAL CONDITIONS
C      T -- STARTING POINT OF INTEGRATION , MUST BE A VARIABLE
C      TOUT -- OUTPUT POINT AT WHICH SOLUTION IS DESIRED.
C            T=TOUT IS ALLOWED ON THE FIRST CALL ONLY, IN WHICH CASE
C            RKF45 RETURNS WITH IFLAG=2 IF CONTINUATION IS POSSIBLE.
C      RELERR,ABSERR -- RELATIVE AND ABSOLUTE LOCAL ERROR TOLERANCES
C            WHICH MUST BE NON-NEGATIVE. RELERR MUST BE A VARIABLE WHILE
C            ABSERR MAY BE A CONSTANT. THE CODE SHOULD NORMALLY NOT BE
C            USED WITH RELATIVE ERROR CONTROL SMALLER THAN ABOUT 1.E-8 .
C            TO AVOID LIMITING PRECISION DIFFICULTIES THE CODE REQUIRES
C            RELERR TO BE LARGER THAN AN INTERNALLY COMPUTED RELATIVE
C            ERROR PARAMETER WHICH IS MACHINE DEPENDENT. IN PARTICULAR,
C            PURE ABSOLUTE ERROR IS NOT PERMITTED. IF A SMALLER THAN
C            ALLOWABLE VALUE OF RELERR IS ATTEMPTED, RKF45 INCREASES
C            RELERR APPROPRIATELY AND RETURNS CONTROL TO THE USER BEFORE
C            CONTINUING THE INTEGRATION.
C      IFLAG -- +1,-1  INDICATOR TO INITIALIZE THE CODE FOR EACH NEW
C            PROBLEM. NORMAL INPUT IS +1. THE USER SHOULD SET IFLAG=-1
C            ONLY WHEN ONE-STEP INTEGRATOR CONTROL IS ESSENTIAL. IN THIS
C            CASE, RKF45 ATTEMPTS TO ADVANCE THE SOLUTION A SINGLE STEP
C            IN THE DIRECTION OF TOUT EACH TIME IT IS CALLED. SINCE THIS
C            MODE OF OPERATION RESULTS IN EXTRA COMPUTING OVERHEAD, IT
C            SHOULD BE AVOIDED UNLESS NEEDED.
C
C
C   OUTPUT FROM RKF45
C
C      Y(*) -- SOLUTION AT T
C      T -- LAST POINT REACHED IN INTEGRATION.
C      IFLAG = 2 -- INTEGRATION REACHED TOUT. INDICATES SUCCESSFUL RETURN
C                   AND IS THE NORMAL MODE FOR CONTINUING INTEGRATION.
C            =-2 -- A SINGLE SUCCESSFUL STEP IN THE DIRECTION OF TOUT
C                   HAS BEEN TAKEN. NORMAL MODE FOR CONTINUING
C                   INTEGRATION ONE STEP AT A TIME.
C            = 3 -- INTEGRATION WAS NOT COMPLETED BECAUSE RELATIVE ERROR
C                   TOLERANCE WAS TOO SMALL. RELERR HAS BEEN INCREASED
C                   APPROPRIATELY FOR CONTINUING.
C            = 4 -- INTEGRATION WAS NOT COMPLETED BECAUSE MORE THAN
C                   3000 DERIVATIVE EVALUATIONS WERE NEEDED. THIS
C                   IS APPROXIMATELY 500 STEPS.
C            = 5 -- INTEGRATION WAS NOT COMPLETED BECAUSE SOLUTION
C                   VANISHED MAKING A PURE RELATIVE ERROR TEST
C                   IMPOSSIBLE. MUST USE NON-ZERO ABSERR TO CONTINUE.
C                   USING THE ONE-STEP INTEGRATION MODE FOR ONE STEP
C                   IS A GOOD WAY TO PROCEED.
C            = 6 -- INTEGRATION WAS NOT COMPLETED BECAUSE REQUESTED
C                   ACCURACY COULD NOT BE ACHIEVED USING SMALLEST
C                   ALLOWABLE STEPSIZE. USER MUST INCREASE THE ERROR
C                   TOLERANCE BEFORE CONTINUED INTEGRATION CAN BE
C                   ATTEMPTED.
C            = 7 -- IT IS LIKELY THAT RKF45 IS INEFFICIENT FOR SOLVING
C                   THIS PROBLEM. TOO MUCH OUTPUT IS RESTRICTING THE
C                   NATURAL STEPSIZE CHOICE. USE THE ONE-STEP INTEGRATOR
C                   MODE.
C            = 8 -- INVALID INPUT PARAMETERS
C                   THIS INDICATOR OCCURS IF ANY OF THE FOLLOWING IS
C                   SATISFIED -   NEQN .LE. 0
C                                 T=TOUT  AND  IFLAG .NE. +1 OR -1
C                                 RELERR OR ABSERR .LT. 0.
C                                 IFLAG .EQ. 0  OR  .LT. -2  OR  .GT. 8
C      WORK(*),IWORK(*) -- INFORMATION WHICH IS USUALLY OF NO INTEREST
C                   TO THE USER BUT NECESSARY FOR SUBSEQUENT CALLS.
```

```
C                    WORK(1),...,WORK(NEQN) CONTAIN THE FIRST DERIVATIVES
C                    OF THE SOLUTION VECTOR Y AT T. WORK(NEQN+1) CONTAINS
C                    THE STEPSIZE H TO BE ATTEMPTED ON THE NEXT STEP.
C                    IWORK(1) CONTAINS THE DERIVATIVE EVALUATION COUNTER.
C
C
C  SUBSEQUENT CALLS TO RKF45
C
C    SUBROUTINE RKF45 RETURNS WITH ALL INFORMATION NEEDED TO CONTINUE
C    THE INTEGRATION. IF THE INTEGRATION REACHED TOUT, THE USER NEED ONL
C    DEFINE A NEW TOUT AND CALL RKF45 AGAIN. IN THE ONE-STEP INTEGRATOR
C    MODE (IFLAG=-2) THE USER MUST KEEP IN MIND THAT EACH STEP TAKEN IS
C    IN THE DIRECTION OF THE CURRENT TOUT. UPON REACHING TOUT (INDICATED
C    BY CHANGING IFLAG TO 2),THE USER MUST THEN DEFINE A NEW TOUT AND
C    RESET IFLAG TO -2 TO CONTINUE IN THE ONE-STEP INTEGRATOR MODE.
C
C    IF THE INTEGRATION WAS NOT COMPLETED BUT THE USER STILL WANTS TO
C    CONTINUE (IFLAG=3,4 CASES), HE JUST CALLS RKF45 AGAIN. WITH IFLAG=3
C    THE RELERR PARAMETER HAS BEEN ADJUSTED APPROPRIATELY FOR CONTINUING
C    THE INTEGRATION. IN THE CASE OF IFLAG=4 THE FUNCTION COUNTER WILL
C    BE RESET TO 0 AND ANOTHER 3000 FUNCTION EVALUATIONS ARE ALLOWED.
C
C    HOWEVER,IN THE CASE IFLAG=5, THE USER MUST FIRST ALTER THE ERROR
C    CRITERION TO USE A POSITIVE VALUE OF ABSERR BEFORE INTEGRATION CAN
C    PROCEED. IF HE DOES NOT,EXECUTION IS TERMINATED.
C
C    ALSO,IN THE CASE IFLAG=6, IT IS NECESSARY FOR THE USER TO RESET
C    IFLAG TO 2 (OR -2 WHEN THE ONE-STEP INTEGRATION MODE IS BEING USED)
C    AS WELL AS INCREASING EITHER ABSERR,RELERR OR BOTH BEFORE THE
C    INTEGRATION CAN BE CONTINUED. IF THIS IS NOT DONE, EXECUTION WILL
C    BE TERMINATED. THE OCCURRENCE OF IFLAG=6 INDICATES A TROUBLE SPOT
C    (SOLUTION IS CHANGING RAPIDLY,SINGULARITY MAY BE PRESENT) AND IT
C    OFTEN IS INADVISABLE TO CONTINUE.
C
C    IF IFLAG=7 IS ENCOUNTERED, THE USER SHOULD USE THE ONE-STEP
C    INTEGRATION MODE WITH THE STEPSIZE DETERMINED BY THE CODE OR
C    CONSIDER SWITCHING TO THE ADAMS CODES DE/STEP,INTRP. IF THE USER
C    INSISTS UPON CONTINUING THE INTEGRATION WITH RKF45, HE MUST RESET
C    IFLAG TO 2 BEFORE CALLING RKF45 AGAIN. OTHERWISE,EXECUTION WILL BE
C    TERMINATED.
C
C    IF IFLAG=8 IS OBTAINED, INTEGRATION CAN NOT BE CONTINUED UNLESS
C    THE INVALID INPUT PARAMETERS ARE CORRECTED.
C
C    IT SHOULD BE NOTED THAT THE ARRAYS WORK,IWORK CONTAIN INFORMATION
C    REQUIRED FOR SUBSEQUENT INTEGRATION. ACCORDINGLY, WORK AND IWORK
C    SHOULD NOT BE ALTERED.
C
C   WORK - Dimensioned for max NEQN = 500
C
      INTEGER NEQN,IFLAG,IWORK(5)
      DOUBLE PRECISION Y(NEQN),T,TOUT,RELERR,ABSERR,WORK(11000)
C     IF COMPILER CHECKS SUBSCRIPTS, CHANGE WORK(1) TO WORK(3+6*NEQN)
C
      EXTERNAL F
C
      INTEGER K1,K2,K3,K4,K5,K6,K1M
C
C
C     COMPUTE INDICES FOR THE SPLITTING OF THE WORK ARRAY
C
      K1M=NEQN+1
      K1=K1M+1
      K2=K1+NEQN
      K3=K2+NEQN
```

```
      K4=K3+NEQN
      K5=K4+NEQN
      K6=K5+NEQN
C
C     THIS INTERFACING ROUTINE MERELY RELIEVES THE USER OF A LONG
C     CALLING LIST VIA THE SPLITTING APART OF TWO WORKING STORAGE
C     ARRAYS. IF THIS IS NOT COMPATIBLE WITH THE USERS COMPILER,
C     HE MUST USE RKFS DIRECTLY.
C
      CALL RKFS(F,NEQN,Y,T,TOUT,RELERR,ABSERR,IFLAG,WORK(1),WORK(K1M),
     1          WORK(K1),WORK(K2),WORK(K3),WORK(K4),WORK(K5),WORK(K6),
     2          WORK(K6+1),IWORK(1),IWORK(2),IWORK(3),IWORK(4),IWORK(5))
C
      RETURN
      END
      SUBROUTINE RKFS(F,NEQN,Y,T,TOUT,RELERR,ABSERR,IFLAG,YP,H,F1,F2,F3,
     1                F4,F5,SAVRE,SAVAE,NFE,KOP,INIT,JFLAG,KFLAG)
C
C     FEHLBERG FOURTH-FIFTH ORDER RUNGE-KUTTA METHOD
C
C
C     RKFS INTEGRATES A SYSTEM OF FIRST ORDER ORDINARY DIFFERENTIAL
C     EQUATIONS AS DESCRIBED IN THE COMMENTS FOR RKF45 .
C     THE ARRAYS YP,F1,F2,F3,F4,AND F5 (OF DIMENSION AT LEAST NEQN) AND
C     THE VARIABLES H,SAVRE,SAVAE,NFE,KOP,INIT,JFLAG,AND KFLAG ARE USED
C     INTERNALLY BY THE CODE AND APPEAR IN THE CALL LIST TO ELIMINATE
C     LOCAL RETENTION OF VARIABLES BETWEEN CALLS. ACCORDINGLY, THEY
C     SHOULD NOT BE ALTERED. ITEMS OF POSSIBLE INTEREST ARE
C         YP - DERIVATIVE OF SOLUTION VECTOR AT T
C         H  - AN APPROPRIATE STEPSIZE TO BE USED FOR THE NEXT STEP
C         NFE- COUNTER ON THE NUMBER OF DERIVATIVE FUNCTION EVALUATIONS
C
C
      LOGICAL HFAILD,OUTPUT
C
      INTEGER  NEQN,IFLAG,NFE,KOP,INIT,JFLAG,KFLAG
      DOUBLE PRECISION  Y(NEQN),T,TOUT,RELERR,ABSERR,H,YP(NEQN),
     1  F1(NEQN),F2(NEQN),F3(NEQN),F4(NEQN),F5(NEQN),SAVRE,
     2  SAVAE
C
      EXTERNAL F
C
      DOUBLE PRECISION  A,AE,DT,EE,EEOET,ESTTOL,ET,HMIN,REMIN,RER,S,
     1  SCALE,TOL,TOLN,U26,EPSP1,EPS,YPK
C
      INTEGER  K,MAXNFE,MFLAG
C
      DOUBLE PRECISION  DABS,DMAX1,DMIN1,DSIGN
C
C     REMIN IS THE MINIMUM ACCEPTABLE VALUE OF RELERR.  ATTEMPTS
C     TO OBTAIN HIGHER ACCURACY WITH THIS SUBROUTINE ARE USUALLY
C     VERY EXPENSIVE AND OFTEN UNSUCCESSFUL.
C
      DATA REMIN/1.0D-12/
C
C
C     THE EXPENSE IS CONTROLLED BY RESTRICTING THE NUMBER
C     OF FUNCTION EVALUATIONS TO BE APPROXIMATELY MAXNFE.
C     AS SET, THIS CORRESPONDS TO ABOUT 500 STEPS.
C
      DATA MAXNFE/2000000/
C
C
C     CHECK INPUT PARAMETERS
C
```

```
C
      IF (NEQN .LT. 1) GO TO 10
      IF ((RELERR .LT. 0.0D0)  .OR.  (ABSERR .LT. 0.0D0)) GO TO 10
      MFLAG=IABS(IFLAG)
      IF ((MFLAG .EQ. 0) .OR. (MFLAG .GT. 8)) GO TO 10
      IF (MFLAG .NE. 1) GO TO 20
C
C     FIRST CALL, COMPUTE MACHINE EPSILON
C
      EPS = 1.0D0
    5 EPS = EPS/2.0D0
      EPSP1 = EPS + 1.0D0
      IF (EPSP1 .GT. 1.0D0) GO TO 5
      U26 = 26.0D0*EPS
      GO TO 50
C
C     INVALID INPUT
   10 IFLAG=8
      RETURN
C
C     CHECK CONTINUATION POSSIBILITIES
C
   20 IF ((T .EQ. TOUT) .AND. (KFLAG .NE. 3)) GO TO 10
      IF (MFLAG .NE. 2) GO TO 25
C
C     IFLAG = +2 OR -2
      IF ((KFLAG .EQ. 3) .OR. (INIT .EQ. 0)) GO TO 45
      IF (KFLAG .EQ. 4) GO TO 40
      IF ((KFLAG .EQ. 5)  .AND.  (ABSERR .EQ. 0.0D0)) GO TO 30
      IF ((KFLAG .EQ. 6)  .AND.  (RELERR .LE. SAVRE)  .AND.
     1    (ABSERR .LE. SAVAE)) GO TO 30
      GO TO 50
C
C     IFLAG = 3,4,5,6,7 OR 8
   25 IF (IFLAG .EQ. 3) GO TO 45
      IF (IFLAG .EQ. 4) GO TO 40
      IF ((IFLAG .EQ. 5) .AND. (ABSERR .GT. 0.0D0)) GO TO 45
C
C     INTEGRATION CANNOT BE CONTINUED SINCE USER DID NOT RESPOND TO
C     THE INSTRUCTIONS PERTAINING TO IFLAG=5,6,7 OR 8
   30 STOP
C
C     RESET FUNCTION EVALUATION COUNTER
   40 NFE=0
      IF (MFLAG .EQ. 2) GO TO 50
C
C     RESET FLAG VALUE FROM PREVIOUS CALL
   45 IFLAG=JFLAG
      IF (KFLAG .EQ. 3) MFLAG=IABS(IFLAG)
C
C     SAVE INPUT IFLAG AND SET CONTINUATION FLAG VALUE FOR SUBSEQUENT
C     INPUT CHECKING.
   50 JFLAG=IFLAG
      KFLAG=0
C
C     SAVE RELERR AND ABSERR FOR CHECKING INPUT ON SUBSEQUENT CALLS
      SAVRE=RELERR
      SAVAE=ABSERR
C
C     RESTRICT RELATIVE ERROR TOLERANCE TO BE AT LEAST AS LARGE AS
C     2*EPS+REMIN TO AVOID LIMITING PRECISION DIFFICULTIES ARISING
C     FROM IMPOSSIBLE ACCURACY REQUESTS
C
      RER=2.0D0*EPS+REMIN
      IF (RELERR .GE. RER) GO TO 55
```

Program RKF45.FOR

```
C
C     RELATIVE ERROR TOLERANCE TOO SMALL
      RELERR=RER
      IFLAG=3
      KFLAG=3
      RETURN
C
   55 DT=TOUT-T
C
      IF (MFLAG .EQ. 1) GO TO 60
      IF (INIT .EQ. 0) GO TO 65
      GO TO 80
C
C     INITIALIZATION --
C                          SET INITIALIZATION COMPLETION INDICATOR,INIT
C                          SET INDICATOR FOR TOO MANY OUTPUT POINTS,KOP
C                          EVALUATE INITIAL DERIVATIVES
C                          SET COUNTER FOR FUNCTION EVALUATIONS,NFE
C                          ESTIMATE STARTING STEPSIZE
C
   60 INIT=0
      KOP=0
C
      A=T
      CALL F(A,Y,YP)
      NFE=1
      IF (T .NE. TOUT) GO TO 65
      IFLAG=2
      RETURN
C
C
   65 INIT=1
      H=DABS(DT)
      TOLN=0.
      DO 70 K=1,NEQN
        TOL=RELERR*DABS(Y(K))+ABSERR
        IF (TOL .LE. 0.) GO TO 70
        TOLN=TOL
        YPK=DABS(YP(K))
        IF (YPK*H**5 .GT. TOL) H=(TOL/YPK)**0.2D0
   70 CONTINUE
      IF (TOLN .LE. 0.0D0) H=0.0D0
      H=DMAX1(H,U26*DMAX1(DABS(T),DABS(DT)))
      JFLAG=ISIGN(2,IFLAG)
C
C
C     SET STEPSIZE FOR INTEGRATION IN THE DIRECTION FROM T TO TOUT
C
   80 H=DSIGN(H,DT)
C
C     TEST TO SEE IF RKF45 IS BEING SEVERELY IMPACTED BY TOO MANY
C     OUTPUT POINTS
C
      IF (DABS(H) .GE. 2.0D0*DABS(DT)) KOP=KOP+1
      IF (KOP .NE. 100) GO TO 85
C
C     UNNECESSARY FREQUENCY OF OUTPUT
      KOP=0
      IFLAG=7
      RETURN
C
   85 IF (DABS(DT) .GT. U26*DABS(T)) GO TO 95
C
C     IF TOO CLOSE TO OUTPUT POINT,EXTRAPOLATE AND RETURN
C
```

```
      DO 90 K=1,NEQN
   90   Y(K)=Y(K)+DT*YP(K)
      A=TOUT
      CALL F(A,Y,YP)
      NFE=NFE+1
      GO TO 300
C
C
C     INITIALIZE OUTPUT POINT INDICATOR
C
   95 OUTPUT= .FALSE.
C
C     TO AVOID PREMATURE UNDERFLOW IN THE ERROR TOLERANCE FUNCTION,
C     SCALE THE ERROR TOLERANCES
C
      SCALE=2.0D0/RELERR
      AE=SCALE*ABSERR
C
C
C     STEP BY STEP INTEGRATION
C
  100 HFAILD= .FALSE.
C
C     SET SMALLEST ALLOWABLE STEPSIZE
C
      HMIN=U26*DABS(T)
C
C     ADJUST STEPSIZE IF NECESSARY TO HIT THE OUTPUT POINT.
C     LOOK AHEAD TWO STEPS TO AVOID DRASTIC CHANGES IN THE STEPSIZE AND
C     THUS LESSEN THE IMPACT OF OUTPUT POINTS ON THE CODE.
C
      DT=TOUT-T
      IF (DABS(DT) .GE. 2.0D0*DABS(H)) GO TO 200
      IF (DABS(DT) .GT. DABS(H)) GO TO 150
C
C     THE NEXT SUCCESSFUL STEP WILL COMPLETE THE INTEGRATION TO THE
C     OUTPUT POINT
C
      OUTPUT= .TRUE.
      H=DT
      GO TO 200
C
  150 H=0.5D0*DT
C
C
C
C     CORE INTEGRATOR FOR TAKING A SINGLE STEP
C
C     THE TOLERANCES HAVE BEEN SCALED TO AVOID PREMATURE UNDERFLOW IN
C     COMPUTING THE ERROR TOLERANCE FUNCTION ET.
C     TO AVOID PROBLEMS WITH ZERO CROSSINGS,RELATIVE ERROR IS MEASURED
C     USING THE AVERAGE OF THE MAGNITUDES OF THE SOLUTION AT THE
C     BEGINNING AND END OF A STEP.
C     THE ERROR ESTIMATE FORMULA HAS BEEN GROUPED TO CONTROL LOSS OF
C     SIGNIFICANCE.
C     TO DISTINGUISH THE VARIOUS ARGUMENTS, H IS NOT PERMITTED
C     TO BECOME SMALLER THAN 26 UNITS OF ROUNDOFF IN T.
C     PRACTICAL LIMITS ON THE CHANGE IN THE STEPSIZE ARE ENFORCED TO
C     SMOOTH THE STEPSIZE SELECTION PROCESS AND TO AVOID EXCESSIVE
C     CHATTERING ON PROBLEMS HAVING DISCONTINUITIES.
C     TO PREVENT UNNECESSARY FAILURES, THE CODE USES 9/10 THE STEPSIZE
C     IT ESTIMATES WILL SUCCEED.
C     AFTER A STEP FAILURE, THE STEPSIZE IS NOT ALLOWED TO INCREASE FOR
C     THE NEXT ATTEMPTED STEP. THIS MAKES THE CODE MORE EFFICIENT ON
C     PROBLEMS HAVING DISCONTINUITIES AND MORE EFFECTIVE IN GENERAL
```

```
C     SINCE LOCAL EXTRAPOLATION IS BEING USED AND EXTRA CAUTION SEEMS
C     WARRANTED.
C
C
C     TEST NUMBER OF DERIVATIVE FUNCTION EVALUATIONS.
C     IF OKAY,TRY TO ADVANCE THE INTEGRATION FROM T TO T+H
C
  200 IF (NFE .LE. MAXNFE) GO TO 220
C
C     TOO MUCH WORK
      IFLAG=4
      KFLAG=4
      RETURN
C
C     ADVANCE AN APPROXIMATE SOLUTION OVER ONE STEP OF LENGTH H
C
  220 CALL FEHL(F,NEQN,Y,T,H,YP,F1,F2,F3,F4,F5,F1)
      NFE=NFE+5
C
C     COMPUTE AND TEST ALLOWABLE TOLERANCES VERSUS LOCAL ERROR ESTIMATES
C     AND REMOVE SCALING OF TOLERANCES. NOTE THAT RELATIVE ERROR IS
C     MEASURED WITH RESPECT TO THE AVERAGE OF THE MAGNITUDES OF THE
C     SOLUTION AT THE BEGINNING AND END OF THE STEP.
C
      EEOET=0.0D0
      DO 250 K=1,NEQN
        ET=DABS(Y(K))+DABS(F1(K))+AE
        IF (ET .GT. 0.0D0) GO TO 240
C
C       INAPPROPRIATE ERROR TOLERANCE
        IFLAG=5
        RETURN
C
  240   EE=DABS((-2090.0D0*YP(K)+(21970.0D0*F3(K)-15048.0D0*F4(K)))+
     1                          (22528.0D0*F2(K)-27360.0D0*F5(K)))
  250   EEOET=DMAX1(EEOET,EE/ET)
C
      ESTTOL=DABS(H)*EEOET*SCALE/752400.0D0
C
      IF (ESTTOL .LE. 1.0D0) GO TO 260
C
C
C     UNSUCCESSFUL STEP
C                        REDUCE THE STEPSIZE , TRY AGAIN
C                        THE DECREASE IS LIMITED TO A FACTOR OF 1/10
C
      HFAILD= .TRUE.
      OUTPUT= .FALSE.
      S=0.1D0
      IF (ESTTOL .LT. 59049.0D0) S=0.9D0/ESTTOL**0.2D0
      H=S*H
      IF (DABS(H) .GT. HMIN) GO TO 200
C
C     REQUESTED ERROR UNATTAINABLE AT SMALLEST ALLOWABLE STEPSIZE
      IFLAG=6
      KFLAG=6
      RETURN
C
C
C     SUCCESSFUL STEP
C                         STORE SOLUTION AT T+H
C                         AND EVALUATE DERIVATIVES THERE
C
  260 T=T+H
      DO 270 K=1,NEQN
```

```
  270   Y(K)=F1(K)
      A=T
      CALL F(A,Y,YP)
      NFE=NFE+1
C
C
C                         CHOOSE NEXT STEPSIZE
C                         THE INCREASE IS LIMITED TO A FACTOR OF 5
C                         IF STEP FAILURE HAS JUST OCCURRED, NEXT
C                            STEPSIZE IS NOT ALLOWED TO INCREASE
C
      S=5.0D0
      IF (ESTTOL .GT. 1.889568D-4) S=0.9D0/ESTTOL**0.2D0
      IF (HFAILD) S=DMIN1(S,1.0D0)
      H=DSIGN(DMAX1(S*DABS(H),HMIN),H)
C
C     END OF CORE INTEGRATOR
C
C
C     SHOULD WE TAKE ANOTHER STEP
C
      IF (OUTPUT) GO TO 300
      IF (IFLAG .GT. 0) GO TO 100
C
C
C     INTEGRATION SUCCESSFULLY COMPLETED
C
C     ONE-STEP MODE
      IFLAG=-2
      RETURN
C
C     INTERVAL MODE
  300 T=TOUT
      IFLAG=2
      RETURN
C
      END
      SUBROUTINE FEHL(F,NEQN,Y,T,H,YP,F1,F2,F3,F4,F5,S)
C
C     FEHLBERG FOURTH-FIFTH ORDER RUNGE-KUTTA METHOD
C
C    FEHL INTEGRATES A SYSTEM OF NEQN FIRST ORDER
C    ORDINARY DIFFERENTIAL EQUATIONS OF THE FORM
C             DY(I)/DT=F(T,Y(1),---,Y(NEQN))
C    WHERE THE INITIAL VALUES Y(I) AND THE INITIAL DERIVATIVES
C    YP(I) ARE SPECIFIED AT THE STARTING POINT T. FEHL ADVANCES
C    THE SOLUTION OVER THE FIXED STEP H AND RETURNS
C    THE FIFTH ORDER (SIXTH ORDER ACCURATE LOCALLY) SOLUTION
C    APPROXIMATION AT T+H IN ARRAY S(I).
C    F1,---,F5 ARE ARRAYS OF DIMENSION NEQN WHICH ARE NEEDED
C    FOR INTERNAL STORAGE.
C    THE FORMULAS HAVE BEEN GROUPED TO CONTROL LOSS OF SIGNIFICANCE.
C    FEHL SHOULD BE CALLED WITH AN H NOT SMALLER THAN 13 UNITS OF
C    ROUNDOFF IN T SO THAT THE VARIOUS INDEPENDENT ARGUMENTS CAN BE
C    DISTINGUISHED.
C
C
      INTEGER  NEQN
      DOUBLE PRECISION  Y(NEQN),T,H,YP(NEQN),F1(NEQN),F2(NEQN),
     1  F3(NEQN),F4(NEQN),F5(NEQN),S(NEQN)
C
      DOUBLE PRECISION  CH
      INTEGER  K
C...
      EXTERNAL F
```

```
      CH=H/4.0D0
      DO 221 K=1,NEQN
  221   F5(K)=Y(K)+CH*YP(K)
      CALL F(T+CH,F5,F1)
C
      CH=3.0D0*H/32.0D0
      DO 222 K=1,NEQN
  222   F5(K)=Y(K)+CH*(YP(K)+3.0D0*F1(K))
      CALL F(T+3.0D0*H/8.0D0,F5,F2)
C
      CH=H/2197.0D0
      DO 223 K=1,NEQN
  223   F5(K)=Y(K)+CH*(1932.0D0*YP(K)+(7296.0D0*F2(K)-7200.0D0*F1(K)))
      CALL F(T+12.0D0*H/13.0D0,F5,F3)
C
      CH=H/4104.0D0
      DO 224 K=1,NEQN
  224   F5(K)=Y(K)+CH*((8341.0D0*YP(K)-845.0D0*F3(K))+
     1                              (29440.0D0*F2(K)-32832.0D0*F1(K)))
      CALL F(T+H,F5,F4)
C
      CH=H/20520.0D0
      DO 225 K=1,NEQN
  225   F1(K)=Y(K)+CH*((-6080.0D0*YP(K)+(9295.0D0*F3(K)-
     1          5643.0D0*F4(K)))+(41040.0D0*F1(K)-28352.0D0*F2(K)))
      CALL F(T+H/2.0D0,F1,F5)
C
C     COMPUTE APPROXIMATE SOLUTION AT T+H
C
      CH=H/7618050.0D0
      DO 230 K=1,NEQN
  230   S(K)=Y(K)+CH*((902880.0D0*YP(K)+(3855735.0D0*F3(K)-
     1        1371249.0D0*F4(K)))+(3953664.0D0*F2(K)+
     2        277020.0D0*F5(K)))
C
      RETURN
      END
```

APPENDIX C

Pulmonary Model Computer Programs

Appendix C - 1
Tissue Model - Part 1

PROGRAM TISSFCM

```
C  This program calculates the Fractional Capacity of lung parenchyma
C  for a given Pressure - Fractional Volume curve for a mechanical unit
c  by adding the effects of terminal bronchi closing and opening.

      DIMENSION CDF(100),FVtin(300),FVtex(300),SFVctb(100)

C COMMON STATEMENTS FOR SUBROUTINE IODELCS
      INTEGER INP, OUT, PLT
      INTEGER CRT, LUOUT, LUIN, LUPLT
      CHARACTER*1 BEEP
      CHARACTER*80 ITITLE
      CHARACTER*64 NAINP,NAOUT,NAPLT
      COMMON /DEVS/ LUTRM, LUIN, LUOUT, LUPLT
      COMMON /CURSOR/ BEEP
      COMMON /LABELS/ ITITLE, NAINP, NAOUT, NAPLT

C SETUP THE OUTPUT FILE
       CRT = 0
       CALL IODECLS(0,1,0)

C  CALCULATE THE EXHALATION P-FV CURVE FOR THE MECHANICAL UNIT

C  Define the normal distribution of closing pressures
C
      Pmean =  -0.8 !Pmean is the mean value of the closing pressures
      SIGMA = 0.5   !SIGMA is the Standard Deviation of the distribution
C     SIGMA = 0.8  for the inhalation distribution

C         Define the FV curve for the mechanical unit when
C                   all Terminal Bronchi are open

      Pzero =-5.0
      Phalf = 5.0
      ALPHA = .5/(Phalf-Pzero)
      BETA = (16*ALPHA**3)/27
C
C     FVmu(P) = ALPHA*(P-Pzero) - BETA*(P-Phalf)**3  for P>Phalf
C     FVmu(P) = ALPHA*(P-Pzero)    for P<Phalf
C
C  Divide the FVA curve into NMAX-1 intervals of DPe width, the Pressue Range
C  is 2.5*SIGMA on each side of the mean closing pressure (PM).
C
      Nmax = 21
      Prange = 2*(2.5*SIGMA)
      DPe = Prange/(Nmax-1)
C  The range is bounded by Pmaxe and Pmine.
C
      Pmaxe = Pmean + Prange/2
      Pmine = Pmean - Prange/2
C
C  Define a variable CDF at the end of each interval,
C  CDF = Cumulative Distribution Function of the closing pressures.
C

      WRITE(0,10)
10    FORMAT('                    ')
      WRITE(0,1)
1     FORMAT('        N       Pcc       CDF(N)')
2     FORMAT(5X,I4,4X,F6.2,4X,F6.3)
```

```
      Pcc = Pmaxe
      DO 100 N = Nmax,1,-1
      U = (Pcc - Pmean)/ SIGMA
      CDF(N) = (ERF(U) + 1)/2.0
      WRITE(0,2) N,Pcc,CDF(N)
100   Pcc = Pcc - DPe
C
      PAUSE'PAUSE AFTER THE CALCULATION OF THE CDF'
C
C  Calculate the FVtis in the region where Terminal Bronchi are closing
C
      WRITE(0,3)
 3    FORMAT('   PRESS    FTB OPEN     FVmu    N    FVtex     FVcpi    SFVc
     Xtb  FVotb')
 4    FORMAT(2X,F6.2,2X,F6.2,7X,F6.3,3X,I2,2X,F6.3,2X,F6.3,2X,F6.3,2X,
     xF6.3)
C
      DPhalfe = DPe/2          !Half a pressure increment DPe
      SFVctb(Nmax) = 0 !Summed Fractional Vol of Closed Terminal Bronchi
C
      Nmax1 = Nmax - 1
      DO 500 N = Nmax1,1,-1
      P = Pmine+(N-1)*DPe+DPhalfe       !P = average pressure in the nth interval
      FVmu = ALPHA*(P-Pzero)
      IF(P.GT.Phalf) THEN FVmu = FVmu - BETA*(P-Phalf)**3
C
      FVcpi = FVmu*(CDF(N+1) - CDF(N))!Fractional volume of gas that Closes
C                               during this Pressure Increment

      SFVctb(N) = SFVctb(N+1) + FVcpi!Summed Fractional Volume of all acini
C                              with Closed Terminal Bronchi
C
      FVotb = FVmu * CDF(N)           !Fractional Volume contribution of
C                              acini with Open Terminal Bronchi
C
      FVtex(N) = SFVctb(N) + FVotb     !Fractional Volume of the Tissue
C
500   WRITE(0,4) P,CDF(N),FVmu,N,FVtex(N),FVcpi,SFVctb(N),FVotb
      PAUSE' PAUSE AFTER OUTPUT OF THE FVtis curve where TB are closed'
C
C Complete FVtex curve for pressures where all terminal bronchi are open
C
      N= Nmax
700   P=Pmine +(N-1)*DPe +DPhalfe
      FVmu = ALPHA*(P-Pzero)
      IF(P.GT.Phalf) FVmu = FVmu -  BETA*(P-Phalf)**3
      FVtex(N) = FVmu
      IF(FVmu.GT.0.9999) GO TO 1000
      N = N+1
      GO TO 700
C
1000  Nmaxte = N      !#+1 of DPe increments in the FVtex curve
C
      J=0
      DO 1200 N=1,Nmaxte
      J = J+1
      IF(J.LT.10) GO TO 1100
      J=0
      PAUSE 'PAUSE DURING PRINT'
1100  P= Pmine + N*DPe - DPhalfe
1200  WRITE(0,9) N,P,FVtex(N)
 9    FORMAT(5X,I3,5X,F6.2,5X,F6.3)

C
C  Calculate the inhalation FVtis curve (FVtin) for a VC inhalation.
C  The inhalation curve for the mechanical unit has the same shape
C  as the exhalation curve
C  and all the P's are increased by an equal amount of hysteresis,
C  DPhys. The opening sequence is taken to be the opposite of the
```

```
C  closing sequence; the first terminal bronchi to close are the last
C  to open.

      SIGMA=0.8
      Pmean=Pmean+5.0
      DPhys=4.0
      Pzero=Pzero+DPhys
      Phalf=Phalf+DPhys

      Prange=2*(2.5*SIGMA)
      DPi=Prange/(Nmax-1)
      DPhalfi=DPi/2.0
      Pmaxi=Pmean+Prange/2
      Pmini=Pmean-Prange/2

      FVtin(1)=FVtex(1)

      WRITE(0,1450)
 1450 FORMAT('   PRESS    FTB OPEN     FVmu   N    FVtin     FVcpi    SFVc
     Xtb  FVotb')

      DO  1500 N=2,Nmax
      P = Pmini+(N-1)*DPi-DPhalfi  !P = average pressure in the nth interval
      FVmu = ALPHA*(P-Pzero)
      IF(P.GT.Phalf) THEN FVmu = FVmu - BETA*(P-Phalf)**3
      SFVotb=CDF(N)*FVmu         !Summed FV of all open TB
      FVtin(N)=SFVotb+SFVctb(N)   !summed FV aof all open and closed TB
 1500 WRITE(0,4) P,CDF(N),FVmu,N,FVtin(N),FVcpi,SFVctb(N),SFVotb
      PAUSE' Pause after output of FVtin curve where all TB have just
     x opened'
C
C Complete FVtin curve for pressures where all terminal bronchi are open
C
      N= Nmax
 1700 P=Pmini +(N-1)*DPi +DPhalfi
      FVmu = ALPHA*(P-Pzero)
      IF(P.GT.Phalf) FVmu = FVmu -  BETA*(P-Phalf)**3
      FVtin(N) = FVmu
      IF(FVmu.GT.0.9999) GO TO 1720
      N = N+1
      GO TO 1700
C
 1720 Nmaxti=N
      FVtin(Nmaxti)=1.0
      Pmaxti=P
C
      J=0
      DO 1800 N=1,Nmaxti
      J = J+1
      IF(J.LT.10) GO TO 1750
      J=0
      PAUSE 'PAUSE DURING PRINT'
1750  P= Pmini + N*DPi - DPhalfi
1800  WRITE(0,9) N,P,FVtin(N)

C  INTERPOLATE THE INHALATION CURVE TO SAME NMAX AS THE EXHALATION CURVE
      CALL INTERP(FVtin,Nmaxti,DPi,Nmaxte,DPe)

C PRINT THE DATA TO AN OUTPUT FILE

 15   FORMAT (1X,I4,3F9.2)
 16   FORMAT(1X,F7.3,1X,F7.3)

      Pminti=Pmini+DPhalfe
      Pmaxti=Pminti+(Nmaxte-1)*DPe
```

```
      WRITE(LUOUT,15) Nmaxte,Pminti,Pmaxti,DPhys

      DO 1900 N=1,Nmaxte
      P= Pmini + N*DPe - DPhalfe
1900  WRITE(LUOUT,16) P,FVtin(N)

      DO 2000 N=1,Nmaxte
      P= Pmine+ N*DPe - DPhalfe
 2000 WRITE(LUOUT,16) P,FVtex(N)
C
      END
C_______________________________________________________________
C
C  THE ERROR FUNCTION
      FUNCTION ERF(X)
      IF(X.LT.0.)THEN
        ERF=-GAMMP(.5,X**2)
      ELSE
        ERF=GAMMP(.5,X**2)
      ENDIF
      RETURN
      END
C_______________________________________________________________
C
C THE GAMMA FUNCTION
      FUNCTION GAMMP(A,X)
      IF(X.LT.0..OR.A.LE.0.)PAUSE
      IF(X.LT.A+1.)THEN
      CALL  GSER(GAMSER,A,X,GLN)
        GAMMP=GAMSER
      ELSE
        CALL GCF(GAMMCF,A,X,GLN)
        GAMMP=1.-GAMMCF
      ENDIF
      RETURN
      END
C_______________________________________________________________
C
      SUBROUTINE GSER(GAMSER,A,X,GLN)
      PARAMETER (ITMAX=100, EPS=3.E-7)
      GLN = GAMMLN(A)
      IF(X.LE.0.)THEN
        IF(X.LT.0.)PAUSE
        GAMSER=0.
        RETURN
      ENDIF
      AP=A
      SUM=1./A
      DEL=SUM
      DO 11 N=1,ITMAX
        AP=AP+1.
        DEL=DEL*X/AP
        SUM=SUM+DEL
        IF(ABS(DEL).LT.ABS(SUM)*EPS)GO TO 1
11    CONTINUE
      PAUSE 'A too large, ITMAX too small'
1     GAMSER = SUM*EXP(-X+A*LOG(X)-GLN)
      RETURN
      END
C_______________________________________________________________
C
      SUBROUTINE GCF(GAMMCF,A,X,GLN)
      PARAMETER (ITMAX=100,EPS=3.E-7)
      GLN=GAMMLN(A)
      GOLD=0.
```

Program TISSFCM.FOR

```
      A0=1.
      A1=X
      B0=0.
      B1=1.
      FAC=1.
      DO 11 N=1,ITMAX
        AN=FLOAT(N)
        ANA=AN-A
        A0=(A1+A0*ANA)*FAC
        B0=(B1+B0*ANA)*FAC
        ANF=AN*FAC
        A1=X*A0+ANF*A1
        B1=X*B0+ANF*B1
        IF(A1.NE.0.)THEN
          FAC=1./A1
          G=B1*FAC
          IF(ABS((G-GOLD)/G).LT.EPS)GO TO 1
          GOLD=G
        ENDIF
11    CONTINUE
      PAUSE 'A too large, ITMAX too small'
1     GAMMCF=EXP(-X+A*ALOG(X)-GLN)*G
      RETURN
      END
C_____________________________________________________________
C
      FUNCTION GAMMLN(XX)
      REAL*8 COF(6),STP,HALF,ONE,FPF,X,TMP,SER
      DATA COF,STP/76.18009173D0,-86.50532033D0,24.01409822D0,
     *    -1.231739516D0,.120858003D-2,-.536382D-5,2.50662827465D0/
      DATA HALF,ONE,FPF/0.5D0,1.0D0,5.5D0/
      X=XX-ONE
      TMP=X+FPF
      TMP=(X+HALF)*LOG(TMP)-TMP
      SER=ONE
      DO 11 J=1,6
        X=X+ONE
        SER=SER+COF(J)/X
11    CONTINUE
      GAMMLN=TMP+LOG(STP*SER)
      RETURN
      END
C_____________________________________________________________
C
      SUBROUTINE IODECLS(INP, OUT, PLT)
      INTEGER INP, OUT, PLT
      CHARACTER*1 BEEP
      CHARACTER*80 ITITLE
      CHARACTER*64 NAINP,NAOUT,NAPLT
      LOGICAL IEXIST
      COMMON /DEVS/ LUTRM, LUIN, LUOUT, LUPLT
      COMMON /CURSOR/ BEEP
      COMMON /LABELS/ ITITLE, NAINP, NAOUT, NAPLT
      QUOTE = '"'
      LUTRM = 0
      LUIN = 1
      LUOUT = 2
      LUPLT = 3
C
C    TERMINAL:  LOGICAL UNIT 0
C       INPUT:  LOGICAL UNIT 1  (DATA FILE)
C      OUTPUT:  LOGICAL UNIT 2
C   PLOT FILE:  LOGICAL UNIT 3
C
      IEXIST =.FALSE.
```

```
      IF (INP .NE. 0) THEN
         WRITE(LUTRM,'(//,5X,''ENTER INPUT FILE NAME.'')')
    5    READ(LUTRM,'(A64)') NAINP
         INQUIRE(FILE=NAINP,EXIST=IEXIST)
         IF (IEXIST) THEN
            OPEN(LUIN,FILE=NAINP,STATUS='OLD')
            WRITE(LUTRM,'(//,5X,''INPUT FILE NAME: '',A64)') NAINP
          ELSE
            WRITE(LUTRM,'(5X,''*** FILE NAME:  '',A64,/,5X,A1,
     *      ''NOT FOUND'')')  NAINP,BEEP
            WRITE(LUTRM,'(/,5X,''ENTER A NEW FILE NAME.'')')
            GOTO 5
          END IF
      END IF
      IEXIST = .FALSE.
      IF (OUT .NE. 0) THEN
          WRITE(LUTRM,'(/,5X,''ENTER OUTPUT FILE NAME.'')')
   10     READ(LUTRM,'(A64)') NAOUT
          INQUIRE(FILE=NAOUT,EXIST=IEXIST)
         IF(IEXIST) THEN
          WRITE(LUTRM,'(5X,''*** FILE NAME:  '',A64,/,5X,A1,
     *    ''ALREADY EXISTS.  DO YOU WISH TO OVERWRITE IT? (Y/N)'')')
     *    NAOUT,BEEP
          READ(LUTRM,'(A1)') ANS
          IF(ANS .EQ. 'Y' .OR. ANS .EQ. 'y') THEN
             OPEN(LUOUT,FILE=NAOUT,IOSTAT=IERR,STATUS='OLD')
             REWIND LUOUT
          ELSE
             WRITE(LUTRM,'(/,5X,''ENTER A NEW FILE NAME'')')
             GOTO 10
          END IF
         ELSE
          OPEN(LUOUT,FILE=NAOUT,IOSTAT=IERR,STATUS='NEW')
         END IF
         WRITE(LUTRM,'(//,5X,''OUTPUT FILE NAME: '',A64)') NAOUT
         WRITE(LUTRM,'(/,5X,''DO YOU WISH TO WRITE A TITLE LINE ON THE'',
     *   '' OUTPUT FILE? (Y/N)'')')
         READ(LUTRM,'(A1)') ANS
         IF(ANS .EQ. 'Y' .OR. ANS .EQ. 'y') THEN
            WRITE(LUTRM,'(/,5X,''ENTER A TITLE LINE.'')')
            READ(LUTRM,'(A80)') ITITLE
            WRITE(LUOUT,'(A1,A80,A1)') QUOTE,ITITLE,QUOTE
          END IF
       END IF
CFF
       IEXIST = .FALSE.
       IF (PLT .NE. 0) THEN
          WRITE(LUTRM,'(/,5X,''ENTER OUTPUT PLOT FILE NAME.'')')
   15     READ(LUTRM,'(A64)') NAPLT
          INQUIRE(FILE=NAPLT,EXIST=IEXIST)
         IF(IEXIST) THEN
          WRITE(LUTRM,'(5X,''*** FILE NAME:  '',A64,/,5X,A1,
     *    ''ALREADY EXISTS.  DO YOU WISH TO OVERWRITE IT? (Y/N)'')')
     *    NAPLT,BEEP
          READ(LUTRM,'(A1)') ANS
          IF(ANS .EQ. 'Y' .OR. ANS .EQ. 'y') THEN
             OPEN(LUPLT,FILE=NAPL~,IOSTAT=IERR,STATUS='OLD')
             REWIND LUPLT
          ELSE
             WRITE(LUTRM,'(/,5X,''ENTER A NEW FILE NAME'')')
             GOTO 15
          END IF
         ELSE
          OPEN(LUPLT,FILE=NAPLT,IOSTAT=IERR,STATUS='NEW')
         END IF
```

```
        WRITE(LUTRM,'(//,5X,''PLOT FILE NAME: '',A64)') NAPLT
      END IF
      RETURN
      END
c-------------------------------------------------------------------------
        SUBROUTINE INTERP(Y,IMAX1,DX1,IMAX2,DX2)
      DIMENSIONY(211), X1(211), X2(211), Y2(211)
      DO 10 I1 = 1,IMAX1
         X1(I1) = (I1 -1)*DX1
   10    CONTINUE
      DO 20 I2 = 1,IMAX2
         X2(I2) = (I2-1)*DX2
   20    CONTINUE
      DO 30 I2 = 1,IMAX2
         I1 = X2(I2)/DX1 + DX1/2.0
         I1 = I1 +1
         IF(I1 .EQ. 1) I1 = 2
         IF(I1 .EQ. IMAX1) I1 = IMAX1 - 1
         U = (X2(I2) - X1(I1))/DX1
         D1 = Y(I1+1) - Y(I1-1)
         D2 = Y(I1+1) - 2.0*Y(I1) + Y(I1-1)
         Y2(I2) = Y(I1) + 0.5*D1*U + 0.5*D2*U**2
   30    CONTINUE
      DO 40 I2 = 1,IMAX2
         Y(I2) = Y2(I2)
   40    CONTINUE
      RETURN
      END
```

Appendix C - 2
Tissue Model - Part 2

PROGRAM REGPVCUR

```
C  THIS PROGRAM DETERMINES THE REGIONAL P-V CURVES FOR THE REGIONS LOCATED
C       AT DIFFERENT VERTICAL POSITIONS
      DIMENSION V(4,200),V0(10),ZQ(10),DV(10),VGAS(10),Ppar(2,201),
     *     POSN(20), PCG(4,200), RFC(4,200), Pst(2,4,200),DPGRAV0(10)
      COMMON A,B,NMAX,NMAX1,CHECKV,DNSTY,IMAX, IMAX1, PSTMIN,
     * PSTMAX, EBAR, VE(101), Z(101), ZI(101), Z0(101), SMASS(101),
     *     P(101),P0(101),DENS(101),E(201),CHECKP,DP,XMASS(101),
     * VETIS(201),VMV(101),VLLUNG(301),VE0(51),VETV(51),
     * CVG3,FI(51),PMIN,TVE,XAZMAX,ZMAX,SA(101),TSMASS(101),
     * BLMASS(101),H(101),PCAP(101),HRTPOS,H0,PSI,
     *     PART,K,ZMAX2,FC(2,201)
      INTEGER R
      INTEGER INP, OUT, PLT
      INTEGER CRT, LUOUT, LUIN, LUPLT
      CHARACTER*1 BEEP, ANS, QUOTE
      CHARACTER*80 ITITLE
      CHARACTER*64 NAINP,NAOUT,NAPLT
      LOGICAL IEXIST
      COMMON /DEVS/LUTRM, LUIN, LUOUT, LUPLT
      COMMON /CURSOR/ BEEP
      COMMON /LABELS/ITITLE,NAINP,NAOUT,NAPLT

    1 FORMAT(I2,2F8.2)
    2 FORMAT(1X,'FC(Jstatus,I) ',10F8.4)
    3 FORMAT(6F20.3)
    4 FORMAT(10X,I3,4F18.3)
    5 FORMAT(5X,I4,2F10.3)
    6 FORMAT(1X,'PMIN PSTMAX DPHYS IM',3F8.3,I3)
    7 FORMAT('IPOS = ',I3)
    8 FORMAT(' VOLUME INCREMENT = ',I3)
1101  FORMAT(3X,'K = ',I3,3X,' LUNG VOL = ',F9.2)
C
C  ISTAT-1 IS A STATIC N2 WASHIN-WASHOUT
C  ISTAT = 2 IS A STUDY OF THE STATIC VOLUME CHANGE OF FOUR
C  REGIONS OF EQUAL MASS
C
      ISTAT = 2
C     ISTAT = 1
C
C  VETIS(I) IS READ IN AS THE GAS FRACTION OF CAPACITY AND MUST BE
C  CONVERTED TO THE VOLUME EXPANSION OF THE TISSUE,BETIS(I).
C
      CRT = 0

C  READ THE DATA FILE WITH THE PARENCHYMAL INHALATION AND EXHALATION
C  P-FV CURVES THAT HAVE BEEN GENERATED IN PROGRAM TISSFCM.  THESE
C  CURVES INCLUDE THE AFFECTS OF TERMINAL BRONCHI CLOSING AND OPENING
C  ON THE PARENCHYMAL ELASTIC CHARACTERISTICS.
C

      CALL IODECLS(1, 0, 0)
  15  FORMAT(1X,I4,3F9.2)
  16  FORMAT(1X,F7.3,1X,F7.3)

      READ (LUIN,15) IM,Pmin,Pstmax,DPhys
C      WRITE(CRT,15) IM,Pmin,Pstmax,DPhys
C NOTE: Pstmin IS DEFINED IN SUBROUTINE VARYMNV

C       PAUSE 'PAUSE DURING THE READING OF THE DATA'
C
```

Program REGPVCUR.FOR

```
C       JSTATUS=1 INDICATES THE INHALATION P-FV CURVE
C       JSTATUS=2 INDICATES THE EXHALATION P-FV CURVE
C
      Jstatus=1
      DO 18 N=1,IM
      READ(LUIN,16) Ppar(Jstatus,N),FC(Jstatus,N) !Inhalation Ppar-FV curve
 18   CONTINUE
C
      Jstatus=2
      DO 19 N=1,IM
      READ(LUIN,16) Ppar(Jstatus,N),FC(Jstatus,N) !Exhalation Ppar-FV curve
 19   CONTINUE
C
C     DO 40 Jstatus=1,2
C     NC=0
C     DO 40 N=1,IM
C     NC=NC+1
C     IF(NC.LT.22) GOTO 40
C     NC=0
C     PAUSE'PAUSE TO LOOK AT Ppar AND FC FROM TISPFV'
C   40     WRITE(CRT,16) Ppar(Jstatus,N),FC(Jstatus,N)
C
C   ANATOMICAL DEAD SPACE [ML]
C
      ADSP = 160.00
      PDS2 = ADSP/2.0
      PDS4 = ADSP/4.0
      IMAX = 201
      IMAX1 = IMAX - 1
C
C  EXPANSION INFORMATION
C
      ALPHA = 0.20
      BETA = 0.30
      GAMMA = 0.50
C
C  LUNG VOLUME, MASS AND GEOMETRY INFORMATION
C  VMAX AND AMV REFER TO THE VOLUME OF TISSUE AND GAS.  RV, FRC
C  AND TLC REFER TO MEASURED GAS VOLUMES.
C
C   RESIDUAL VOLUME [ML]
C
        RV = 2000.0     !NRA
C       RV = 1890.00    !JCC
C       RV = 1000.00    !TEST CASE
C
C   FUNCTIONAL RESIDUAL CAPACITY [ML]
C
        FRC = 3810.00          !NRA
C       FRC = 3400.00          !JCC
C       FRC = 2500.00          !TEST CASE
C
C   TOTAL LUNG CAPACITY [ML]
C
        TLC = 7600.00          !NRA
C       TLC = 6400.00          !JCC
C       TLC = 4000.00          !TEST CASE
C
C  CORRECT VOLUMES FOR ANATOMICAL DEAD SPACE
C
       RV = RV - ADSP
       FRC = FRC - ADSP
       TLC = TLC - ADSP
C
C      GRAMS = MASS OF LUNG = OVERALL DENSITY * TOTAL LUNG CAPACITY;
```

Program REGPVCUR.FOR

```
C         THE DENSITY AT TLC IS ASSUMED TO BE 0.13333gm/cc
C
      GRAMS = TLC*0.13333
C
C     GET VOLUMES AND WEIGHT FOR ONE LUNG
C
      GRAMS = GRAMS/2.0
      RV = RV/2.0
      FRC = FRC/2.0
      TLC = TLC/2.0
      VC=TLC-RV                  !Vital Capacity for a single lung
      VMAX = TLC + GRAMS
      Jstatus=1                  !Inhalation status
      FCMIN = FC(Jstatus,1)

      AMV = FCMIN*TLC + GRAMS !Amount Minimum Volume = smallest air
c                              vol + tissue vol outside chest.
C
C  DENSITY = MASS/VOLUME
C
      DNSTY = GRAMS/AMV
C      WRITE(0,3) DNSTY,AMV,GRAMS
C      PAUSE 'PAUSE AFTER DNSTY,AMV AND GRAMS ARE PUT ON THE SCREEN'

      TC = AMV/1250.
      A0 = 7.00*TC**ALPHA
      B0 = 5.25*TC**BETA
      C0 = 10.50*TC**GAMMA
      THETA = 3.1416*A0*B0/C0**2
C
C  CALCULATE THE TISSUE FC-P RELATION FOR THE SUBJECT.  THE TISSUE
C  FCMIN IS BASED ON THE RATIO RV/T1C
C
      IPOS = 2
C      WRITE(CRT,7) IPOS
C      PAUSE'PAUSE BEFORE YOU ENTER  VRYMNV FOR THE FIRST TIME'
      CALL VRYMNV(FCMIN,TLC,GRAMS,IM,AMV,Jstatus)
      IPOS = 3
C      WRITE(CRT,7) IPOS
C
C  CALCULATION OF ZMAX FOR THE GIVEN SHAPE AND GAS AND TISSUE
C  VOLUME.
C
 65   FORMAT (2X,' ZMAX0 = ',F7.2)
      G1 = 1.0
      G2 = -C0*3.0
      G3 = 0.0
      G4 = 3.0*AMV/THETA
      Z1 = 20.0
      CALL SOLVEZ(G1,G2,G3,G4,Z1,ZMAX0)
      WRITE(CRT,65) ZMAX0
C      PAUSE'PAUSE AFTER YOU WRITE ZMAX0'
C
C   INFORMATION THE BREATH BEING ANALYZED
C
      WRITE(CRT,3) RV, FRC, TLC
C
      VSTART = FCmin*TLC       ! Vstart=lung volume inhalation begins
      VSTOP =  FCmin*TLC       ! Vstop=lung volume exhalation ends
      TIDVOL=TLC*(1.0-FCMIN)   ! TIDal VOLume for this breath
C
C  HEART AND BLOOD VOLUME INFORMATION
C
      PART = 10.00
      HRTPOS = 15.00
```

Program REGPVCUR.FOR

```
      SAPGT = 30000.00
      NO = 0.00000428
      PSI = 0.000000219
C
C  NUMBER OF LUNG SECTIONS
C
      NMAX = 51
      NMAX1 = NMAX - 1
      DZ = ZMAX0/NMAX1
C
C     CONDITIONS WHEN THE LUNG IS AT MINIMUM VOLUME
C
      ZI(1) = 0.0
      Z(1) = 0.0
      XMASS(1) = 0.0
      SMASS(1) = 0.0
      VMV(1) = 0.0
      VE(1) = 0.
      JCOUNT=0
      DO 20 N = 2,NMAX
       JCOUNT=JCOUNT+1
         VE(N) = 1.0
         ZI(N) = (N-1)*DZ
         VMV(N) = THETA * (CO * (ZI(N)**2 - (ZI(N-1))**2) -
     *          (ZI(N)**3 - ZI(N-1)**3)/3.0)
         TSMASS(N) = DNSTY * VMV(N)
         SA(N) = SAPGT*TSMASS(N)
       XMASS(N) = TSMASS(N) + XMASS(N-1)
C      WRITE(0,21)N,VMV(N),TSMASS(N),SA(N),XMASS(N)
       IF(JCOUNT.LT.20) GOTO 20
       JCOUNT=0
C      PAUSE' PAUSE TO LOOK AT N,VMV(N),TSMASS(N),SA(N),XMASS(N)'
   20 CONTINUE
   21 FORMAT(2X,I5,4(2X,F8.3))
C     WRITE(CRT,3) AMV, XMASS(NMAX)
C     PAUSE'PAUSE AFTER VMV PRINT OUT'

       RWEIGHT = GRAMS/4.0       !Tissue weight of a region in gms
       RTLC = TLC/4.0            !Regional gas volume in cc
      DO 25 L = 1, 4
       V(L,1) = AMV/4.0
  25  CONTINUE

C    DETERMINE POSN(J), ODD VALUES OF J DEFINE THE POSITION OF THE CG
C    ON THE MATERIAL COORDINATE SYSTEM, EVEN VALUES DEFINE THE LOWER
C    SURFACE OF EACH REGION ON THE MATERIAL COORDINATE SYSTEM.
C
      FRSIZE=.125
      K=1
      FRMASS=0
      ICOUNT=0

      DO 30 N = 2,NMAX
       FRMASSO=FRMASS
       FRMASS = XMASS(N)/XMASS(NMAX)
       IF(FRMASS.LT.K*FRSIZE) GOTO 26
       POSN(K)=N-1+(K*FRSIZE-FRMASSO)/(FRMASS-FRMASSO)
       K=K+1
  26   CONTINUE
C      WRITE(CRT,5) N-1, XMASS(N), FRMASS
       ICOUNT=ICOUNT+1
       IF(ICOUNT.NE.17) GOTO 30
       ICOUNT=0
C      PAUSE'PAUSE TO LOOK AT THE MASS DISTRIBUTION'
  30  CONTINUE
```

Program REGPVCUR.FOR

```
C       PAUSE'PAUSE TO LOOK AT THE MASS DISTRIBUTION'

        POSN(8)=  NMAX + .000001

        K1= POSN(1)
        K2= POSN(2)
        K3= POSN(3)
        K4= POSN(4)
        K5= POSN(5)
        K6= POSN(6)
        K7= POSN(7)
        K8= POSN(8)
C
C  CALCULATE THE DPGRAV); THE PRESSURE DIFFERENCE BETWEEN THE REGIONAL
C                 CG's AT THE MINIMUM VOLUME
C
        DPGRAVO(1)= (POSN(3)-POSN(1))*DZ*DNSTY
        DPGRAVO(2)= (POSN(5)-POSN(3))*DZ*DNSTY
        DPGRAVO(3)= (POSN(7)-POSN(5))*DZ*DNSTY
C
C       DO 31 K=1,8
C 31    WRITE(CRT,5) K,POSN(K),DPGRAVO(K)
C 32    FORMAT(5X,8I6)
C       PAUSE'PAUSE AFTER THE PRINTING OF THE POSN(K),DPGRAVO'
C
C  ITERATION INFORMATION
C
       CHECKV = 2.0
       CHECKP = 0.01
       DPI = 0.5
       P2 = 4.00
C
C   CALCULATE THE NUMBER OF VOLUME INCREMENTS (DVOL) FOR THIS BREATH.
C   THE LAST ONE IS K End EXhalation (KEEX). THE SIZE OF DVOL IS
C   ESTIMATED THEN ADJUSTED SLIGHTLY SO THAT IT FITS EVENLY INTO THE
C   TIDAL VOLUME (TIDVOL).
C
        DVOL=40.00
        KEINSM1=TIDVOL/DVOL          ! KEINS-1
        DVOL=TIDVOL/KEINSM1          ! Recalculate DVOL
        KEINS=KEINSM1+1              ! Calculate K End INspiration
        KEINSP1=KEINS+1              ! KEINS+1
        KEEX=KEINS+((VSTART+TIDVOL)-VSTOP)/DVOL!Calculate KEEX
        KEEXM1=KEEX-1
        KEEXM2=KEEX-2
        KEEXP1=KEEX+1
 35    FORMAT(4X,5F10.2)
C       WRITE(0,35) VLLUNG(1),VSTART,GRAMS,TIDVOL,DVOL
C       PAUSE'PAUSE AFTER PRINTING VLLUNG(1),VSTART,GRAMS,TIDVOL,DVOL'

       VLLUNG(1) = VSTART + GRAMS

       DO 1020 K = 2,KEINS
 1020  VLLUNG(K) = VLLUNG(K-1) + DVOL
       DO 1025 K=KEINSP1,KEEX
 1025  VLLUNG(K) =VLLUNG(K-1) -DVOL

       DO 1050 L=1,4
 1050  VO(L)=V(L,1)

C
C--------------------MAIN TIME DO LOOP--------------------------------
C
        K=0
 5000   K=K+1
```

```
      IF(K.GT.KEEX) GOTO 500

      WRITE(CRT,1101) K,VLLUNG(K)
C     PAUSE'PAUSE BEFORE THE Kth LUNG VOLUME CALC STARTS'

      IF(K.NE.KEINSP1) GOTO 100
C
C  CONVERT THE EXHALATION P-FV CURVE TO A P-VE (Volume Expansion) CURVE
C
      GOTO 2008
   99 CONTINUE
      Jstatus=2
      Pmin=Pmin-DPhys
      Pstmax=Pstmax-DPhys
      CALL VRYMNV(FCNIN,TLC,GRAMS,IN,AMV,Jstatus)
      P2=P2-DPhys

  100 CONTINUE
        VOL = VLLUNG(K)
        VOLEXP = VOL/AMV
        ZE = VOLEXP**GAMMA
        ZMAX = ZMAX0*ZE
        ZMAX2 = ZMAX + 2.0
        TVE = THETA*VOLEXP**(ALPHA + BETA - 2.0*GAMMA)
        CVG = CO*VOLEXP**GAMMA
        CVG3 = CVG*3
        XAZMAX = TVE*(2.0*CVG*ZMAX - ZMAX**2)
C
C FIRST APPROX FOR DENS(N) AND Z(N) - USE THE VALUES FOR THE LUNG
C IN THE UNWEIGHTED STATE - AS WEST STARTED HIS OUT.
C
        IF(K .EQ. 1) THEN
         DO 110 N = 2,NMAX
             DENS(N) = DNSTY/VOLEXP
             Z(N) = ZE*ZI(N)
  110   CONTINUE
        END IF
      DO 120 N = 2,NMAX
           VEO(N) = VE(N)
           ZO(N) = Z(N)
  120 CONTINUE
        VCALO = VCALC
C
C FOR THE GIVEN P2,DENS,Z AND GEOMETRY (R**2 = A*(1 + EBAR)*2/PI*Z,
C FIND THE VOLUME-VCALC
C
C      PAUSE'PAUSE BEFORE THE FIRST CALL DPDG'
      CALL DPDG(P2, VCALC)
C     WRITE(0,3) P2, VOL, VCALC
        F2 = VOL - VCALC
        IF(F2 .LT. 0.0) THEN
            P3 = P2 - DPI
          ELSE
            P3 = P2 + DPI
        END IF
  130 F1 = F2
        P1 = P2
      P2 = P3
C     PAUSE'PAUSE BEFORE THE SECOND CALL DPDG'
      CALL DPDG(P2,VCALC)
        F2 = VOL - VCALC
      AF2 = ABS(F2)
      WRITE(CRT,51) P1,P2,F1,F2,VOL,VCALC
   51 FORMAT(2X,6F9.2)
      IF(AF2 .LT. CHECKV) GO TO 140
```

Program REGPVCUR.FOR

```
        P3 = (P1*F2 - P2*F1)/(F2-F1)
       GO TO 130
 140   CONTINUE

       ICOUNT=0
 101   FORMAT(2X,I3,3F10.2)
       DO 150 N=1,NMAX
C      WRITE(0,101) N,P(N),VE(N),VMV(N)
       ICOUNT=ICOUNT + 1
       IF(ICOUNT.NE.17) GO TO 150
C      PAUSE'PAUSE TO CHECK N,P(N),VE(N),VMV(N) DATA'
       ICOUNT=0
  150  CONTINUE
C
C CALCULATE THE VOLUME AND THE STATIC RECOIL PRESSURE AT THE CG OF EACH
C          OF THE FOUR REGIONS IN THE DYNAMIC MODEL
C
C
C                        REGION 1
       V(1,K) = 0.0
       DO 220 N = 2,K2
          V(1,K) = V(1,K) + VE(N)*VMV(N)
 220   CONTINUE
       V(1,K) = V(1,K) +(POSN(2)-K2)*VE(K2+1)*VMV(K2+1)
       PCG(1,K)= P(K1) + (POSN(1)-K1)*(P(K1+1)-P(K1))
C
C                        REGION 2

       V(2,K) = (K2+1-POSN(2))*VE(K2+1)*VMV(K2+1)
       DO 230 N = K2+2,K4
          V(2,K) = V(2,K) + VE(N)*VMV(N)
 230     CONTINUE
       V(2,K) = V(2,K) + (POSN(4)-K4)*VE(K4+1)*VMV(K4+1)
       PCG(2,K)= P(K3) + (POSN(3)-K3)*(P(K3+1)-P(K3))
C
C                        REGION 3
       V(3,K) = (K4+1-POSN(4))*VE(K4+1)*VMV(K4+1)
       DO 240 N = K4+2,K6
          V(3,K) = V(3,K) + VE(N)*VMV(N)
 240   CONTINUE
       V(3,K)=V(3,K)+(POSN(6)-K6)*VE(K6+1)*VMV(K6+1)
       PCG(3,K)= P(K5) + (POSN(5)-K5)*(P(K5)-P(K5+1))
C
C                        REGION 4
       V(4,K) = (K6+1-POSN(6))*VE(K6+1)*VMV(K6+1)
       DO 250 N=K6+2,NMAX
          V(4,K) = V(4,K) + VE(N)*VMV(N)
 250   CONTINUE
       PCG(4,K)= P(K7) + (POSN(7)-K7)*(P(K7)-P(K7+1))

       VOLCHG = VCALC - VCALO
C      WRITE(0,4)
 255   FORMAT(2X,I3,5F10.3)
         DO 260 L = 1,4
          VGAS(L) = V(L,K) - RWEIGHT
          DV(L) = V(L,K) -VO(L)
          ZQ(L) = DV(L)/VOLCHG
          WRITE(CRT,255) L,V(L,K),PCG(L,K),VGAS(L), DV(L), ZQ(L)
 260   CONTINUE

       GASVOL = 2.0*(VLLUNG(K) - GRAMS) + ADSP
         DO 265 L = 1,4
          VREG = 2.0*VGAS(1) + PDS4
          VO(L)=V(L,K)
 265     CONTINUE
```

Program REGPVCUR.FOR

```
      GOTO 5000
C----------------------END TO TIME LOOP----------------------------
C
 500 CONTINUE
C
C  PRINT PROGRAM OUTPUT TO A NAMED DATA FILE FOR BY THE DYNamic
C  REGional  flow (Q) PROGRAM (PRG DYNREGQ).  THE DATA WILL REPRESENT
C  THE REGIONAL LUNG CHARACTERISTICS FOR A SPECIFIC SUBJECT.
C
C            NAME THE OUTPUT FILE
C
     CALL IODECLS(0,1,0)
C  RECONSTRUCT RV,FRC,TLC,AMV AND GRAMS FOR THIS SUBJECT
C
     RV=2*RV+ADSP
     FRC=2*FRC+ADSP
     TLC=2*TLC+ADSP
     GRAMS=2*GRAMS
     AMV=2*AMV
C
C  WRITE THE RV,FRCTLC,AND ADSP FOR THIS SUBJECT TO THE NAMED FILE
     WRITE(LUOUT,2005) RV,FRC,TLC,ADSP
C
C  WRITE THE EXPANSION COEFFICIENTS, LUNG VOLUME(GAS AND TISSUE) AT
C  MINIMUM VOLUME AND THE TISSUE WEIGHT OF THE LUNGS.
C
     WRITE(LUOUT,2005)ALPHA,BETA,AMV,GRAMS
C
C  WRITE OUT THE PRESSURE DIFFERENCES BETWEEN THE CG's OF THE FOUR
C  REGIONS WHEN THEY ARE AT MINIMUM VOLUME.
C
     WRITE(LUOUT,2005) (DPGRAVO(I),I=1,3)
C
C WRITE OUT THE EXHALATION P-FV CURVE; FIRST DETERMINE THE REGIONAL Pst-
C FV RELATION AS A FUNCTION OF A CONSTANT DFC.THIS RELATIONSHIP WILL
C BE INPUT INTO PRG DYNREGQ.
C
 2008  CONTINUE
       JMAX=50
       JMAXM1=JMAX-1
       FCMAX=1.0
       DFC=(FCMAX-FCMIN)/JMAXM1
C
 2001 FORMAT(5X,' REGION =',I2)
 2002 FORMAT(3X,2I5,4F10.2)
 2003 FORMAT(I3,3F10.4)
 2004 FORMAT(10F8.4)
 2005 FORMAT(5F10.4)
 2006 FORMAT(3X,I5,4F10.3)
C
     INCREMENT=1
     IF(Jstatus.EQ.2) INCREMENT=-1
     KJBEG=1
     IF(Jstatus.EQ.2) KJBEG=KEEX
     KJEND=KEINS
     IF(Jstatus.EQ.2) KJEND=KEINS
C
C  CONVERT THE REGIONAL GAS VOLUMES INTO REGIONAL GAS FRACTIONS (RFC)
C
     DO 2050 R=1,4
     WRITE(CRT,2001) R
     ICOUNT=0
     DO 2010 KJ=KJBEG,KJEND,INCREMENT
     ICOUNT=ICOUNT+1
     RFC(R,KJ)=(V(R,KJ)-RWEIGHT)/RTLC
```

Program REGPVCUR.FOR

```
      WRITE(CRT,2006) KJ, V(R,KJ), RFC(R,KJ), PCG(R,KJ), RTLC
      IF(ICOUNT.LT.20) GOTO 2010
      PAUSE'PAUSE TO LOOK AT  KJ, V(R,KJ), RFC(R,KJ), PCG(R,KJ), RTLC'
      ICOUNT=0
 2010 CONTINUE
      PAUSE'PAUSE TO LOOK AT  KJ, V(R,KJ), RFC(R,KJ), PCG(R,KJ), RTLC'
C
C  DETERMINE THE Pst AS A FUNCTION OF RFC IN INCREMENTS OF DFC.J IS
C  THE VARIABLE FOR THE NEW Pst-FC DATA FOR PRG DYNREGQ.  KJ IS THE
C  VARIABLE FOR THE REGIONAL Pcg-RFC DATA THAT REPRESENTS THE
c  VOLUME INCREMENTS DURING THE INHALATION AND EXHALATION. THE
C  CURVE INTERPOLATION STARTS AT FCMIN (J=1)AND INCREASES IN DFC
C  INCREMENTS, SO THE INITIAL VALUE FOR KJ, KBEG,DEPENDS ON
C  WHETHER IT IS FOR INHALATION OR EXHALATION.
C
       J=1
       KJ=KJBEG
       ICOUNT=0
 2020  ICOUNT=ICOUNT+1
       FCJ=FCMIN+(J-1)*DFC
       IF(J.EQ.1) GOTO 2035
 2030  IF(RFC(R,KJ).GT.FCJ) GOTO 2040
       IF(KJ.EQ.KJEND) GOTO 2040
       KJ=KJ+INCREMENT
       GOTO 2030
 2035  CONTINUE
C
C  CALCULATE THE PST FOR THE FIRST POINT ON THE PST CURVE (J=1)
C
       KJ1=KJ
       KJ2=KJ+INCREMENT
 2037  IF(RFC(R,KJ2).GT.RFC(R,KJ1)) GOTO 2038
       KJ1=KJ1+INCREMENT
       KJ2=KJ2+INCREMENT
       GOTO 2037
 2038  U=(FCJ-RFC(R,KJ1))/(RFC(R,KJ2)-RFC(R,KJ1))
       PST(Jstatus,R,J)=Pcg(R,KJ1)+U*(Pcg(R,KJ2)-PCG(R,KJ1))
      WRITE(CRT,2002) J,KJ,FCJ,Pst(Jstatus,R,J),PCG(R,KJ),RFC(R,KJ)
       KJ=KJ2
       J=J+1
       GOTO 2020
 2040  U=(FCJ-RFC(R,KJ-1))/(RFC(R,KJ)-RFC(R,KJ-1))
       PST(Jstatus,R,J)=Pcg(R,KJ-1)+U*(Pcg(R,KJ)-Pcg(R,KJ-1))
      WRITE(CRT,2002) J,KJ,FCJ,Pst(Jstatus,R,J),PCG(R,KJ),RFC(R,KJ)
       IF(ICOUNT.LT.22) GOTO 2045
      PAUSE'PAUSE TO SEE J,KJ,FCJ,Pst(Jstatus,R,J),PCG(R,KJ),RFC(R,KJ)'
       ICOUNT=0
 2045  IF(J.EQ.JMAX)GOTO 2048
       J=J+1
       GOTO 2020
 2048  CONTINUE
      PAUSE'PAUSE TO SEE J,KJ,FCJ,Pst(Jstatus,R,J),PCG(R,KJ),RFC(R,KJ)'
 2050  CONTINUE
       IF(JSTATUS.EQ.1) GOTO 99
C
C  WRITE PST DATA TO A NAMED FILE FOR USE IN PROGRAM DYNREGQ
C
      WRITE(LUOUT,2003) JMAX,FCMIN,FCMAX
      DO 2100 R=1,4
      DO 2100 Jstatus=1,2
 2100 WRITE(LUOUT,2004)(Pst(Jstatus,R,J), J=1,JMAX)
    . WRITE(LUOUT,2003) IM,Pmin,Pstmax,DPhys
      WRITE(LUOUT,2004)(FC(1,N),N=1,IM)    !Original data from TISPV
      WRITE(LUOUT,2004)(FC(2,N),N=1,IM)    !Original data from TISPV
      STOP
```

```
      END
C-----------------------------------------------------------------
      SUBROUTINE DPDG(P2,VCALC)
C Subroutine calculates the tissue pressures at the top of each section
c    for the given density using the hydrostatic pressure equation
        COMMON A,B,NMAX,NMAX1,CHECKV,DNSTY,IMAX, IMAX1, PSTMIN,
     * PSTMAX, EBAR, VE(101), Z(101), ZI(101), ZO(101), SMASS(101),
     *     P(101),PO(101),DENS(101),E(201),CHECKP,DP,XMASS(101),
     * VETIS(201),VMV(101),VLLUNG(301),VEO(51),VETV(51),
     * CVG3,FI(51),PMIN,TVE,XAZMAX,ZMAX,SA(101),TSMASS(101),
     * BLMASS(101),H(101),PCAP(101),HRTPOS,HO,PSI,
     *     PART,K,ZMAX2,FC(2,201)
 10   FORMAT(3X,3F12.3)
C      PAUSE'PAUSE BEFORE THE FIRST P(N) CALC IN DPDG'
      P(1) = P2
      DO 20 N = 2,NMAX
       P(N) = P(N-1) - DENS(N)*(Z(N) - Z(N-1))
C     WRITE(0,10) P(N),Z(N),DENS(N)
 20   CONTINUE
C     PAUSE'PAUSE BEFORE THE FIRST CALL CLCVL IN DPDG'
      CALL CLCVL(VCALC)
 25   DO 30 N = 2,NMAX
         PO(N) = P(N)
 30   CONTINUE
C  PAUSE'PAUSE BEFORE THE SECOND P(N) CALC IN DPDG'
      DO 40 N = 2,NMAX
       P(N) = P(N-1) - DENS(N)*(Z(N) - Z(N-1))
C      WRITE(0,10) P(N),Z(N),DENS(N)
 40   CONTINUE
C     PAUSE'PAUSE BEFORE THE SECOND CALL CLCVL'
      CALL CLCVL(VCALC)
      G = PO(NMAX) - P(NMAX)
      AG = ABS(G)
      IF(AG .GT. CHECKP) GO TO 25
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE CLCVL(VCALC)
      COMMON A,B,NMAX,NMAX1,CHECKV,DNSTY,IMAX, IMAX1, PSTMIN,
     * PSTMAX, EBAR, VE(101), Z(101), ZI(101), ZO(101), SMASS(101),
     *     P(101),PO(101),DENS(101),E(201),CHECKP,DP,XMASS(101),
     * VETIS(201),VMV(101),VLLUNG(301),VEO(51),VETV(51),
     * CVG3,FI(51),PMIN,TVE,XAZMAX,ZMAX,SA(101),TSMASS(101),
     * BLMASS(101),H(101),PCAP(101),HRTPOS,HO,PSI,
     *     PART,K,ZMAX2,FC(2,201)

      VCALC = 0.0
      DO 301 N = 2,NMAX
         II = (P(N) - PSTMIN)/DP + 1
         IF(II .LT. 2) GO TO 100
         IF(II. GT. IMAX1) GO TO 200
         U = P(N) - (PSTMIN + DP*(II-1))
       VE(N) = VETIS(II-1) + U*(VETIS(II + 1) - VETIS(II-1))/(2.*DP)
     &           + U**2*(VETIS(II+1) - 2.0*VETIS(II) + VETIS(II-1))/DP**2
         GO TO 300
 100     U = P(N) - PSTMIN
         VE(N) = VETIS(1) + U*(-1.5*VETIS(1) + 2.0*VETIS(2) -
     &           0.5*VETIS(3))/DP
         IF(P(N) .LE. PSTMIN) VE(N) = VETIS(1)
         GO TO 300
 200   U = P(N) - (PSTMAX -DP)      ! P at IMAX1 =PSTMAX-DP
     . VE(N) = VETIS(IMAX1) + U*(VETIS(IMAX) - VETIS(IMAX1))/DP
 300     IF(VE(N) .LT. 1.00) VE(N) = 1.00
         VCALC = VCALC + VE(N)*VMV(N)
 301  CONTINUE
```

Program REGPVCUR.FOR

```
C
C     CALCULATE THE BLOOD VOLUME, BASED ON THE OLD Z POSITION,
C     AND THE DENSITY, BASED ON THE BLOOD MASS AND THE TISSUE EXPANSION
C
 99   FORMAT(2X,I5,6(1X,F8.2))
      JCOUNT=0
      SUM=0
      VE(1) = VE(2)
      DO 310 N = 2,NMAX
       JCOUNT=JCOUNT+1
         PALV = 0.0
         IF(P(N) .LT. 0.0) PALV = -P(N)
         PCAP(N) = PART + ((Z(N) + Z(N-1))/2.0 - HRTPOS)
         PTM = PCAP(N) - PALV
         H(N) = H0 + PSI*PTM
         IF(PTM .LT. 0.0) H(N) = 0.0
         SMASS(N) = TSMASS(N) + BLMASS(N)
       DENS(N) = SMASS(N)/(VMV(N)*(VE(N) + VE(N-1))/2.0 + BLMASS(N))
       SUM=SUM+VE(N)*VMV(N)
C      WRITE(0,99) N,DENS(N),SMASS(N),VMV(N),VE(N),TSMASS(N),SUM
       IF(JCOUNT.LT.20) GOTO 310
C      PAUSE'PAUSE TO LOOK AT N,DENS,SMASS,VMV,VE,TSMASS,CALCVOL'
      JCOUNT=0
 310  CONTINUE

      DO 400 N = 2,NMAX
C
C     CALCULATE THE NEW Z POSITIONS
C
         IF (Z(N-1) .GT. ZMAX2) GO TO 350
         G1 = 1.0
         G2 = -CVG3
         G3 = 0.0
         Z1 = ZO(N)
 315     G4 = CVG3*Z(N-1)**2 - Z(N-1)**3 + 3.0*SMASS(N)/(DENS(N)*TVE)
 320   CALL SOLVEZ(G1,G2,G3,G4,Z1,Z(N))
         IF(Z(N) .GT. 0.0) GO TO 400
         Z1 = Z1 + 1.0
         GO TO 320
 350     Z(N) = Z(N-1) + SMASS(N)/(DENS(N)*XAZMAX)
 400  CONTINUE
      DO 500 N = 2,NMAX
         ZO(N) = Z(N)
 500  CONTINUE
      RETURN
      END
C----------------------------------------------------------------------
      SUBROUTINE VRYMNV(FCMIN,TLC,GRAMS,IM,AMV,JSTATUS)

       COMMON A,B,NMAX,NMAX1,CHECKV,DNSTY,IMAX, IMAX1, PSTMIN,
     *  PSTMAX, EBAR, VE(101), Z(101), ZI(101), ZO(101), SMASS(101),
     *     P(101),PO(101),DENS(101),E(201),CHECKP,DP,XMASS(101),
     *  VETIS(201),VMV(101),VLLUNG(301),VEO(51),VETV(51),
     *  CVG3,FI(51),PMIN,TVE,XAZMAX,ZMAX,SA(101),TSMASS(101),
     *  BLMASS(101),H(101),PCAP(101),HRTPOS,HO,PSI,
     *     PART,K,ZMAX2,FC(2,201)

      INTEGER CRT, OFILE
      COMMON /DEVS/ CRT, LUIN, OFILE, LUPLT

C
C  DETERMINE THE PRESSURE AT WHICH THE VOLUME STARTS TO CHANGE -
C  PSTMIN; AND CHANGE THE FC DATA TO VOLUME EXPANSION (VETIS) DATA
```

Program REGPVCUR.FOR

```
C
  3   FORMAT(5X,I5,4F12.4)
  4   FORMAT(10X,I3,4F12.3)
      DPC = (PSTMAX - PMIN)/(IM-1)
      I = 1
 5    I = I+1
      IF(FC(Jstatus,I) .GT. FC(Jstatus,1)) GO TO 6
      GO TO 5
 6    PSTMIN = PMIN + (I-2)*DPC
C     WRITE(0,3)IM,PSTMIN,PMIN,PSTMAX,DPC
C     PAUSE'PAUSE AFTER PRINTING OF IM,PSTMIN,PMIN,PSTMAX,DPC'
      ILF = I-2
      IMF = IM - ILF
      DO 7 I = 1,IMF
           IXF = I + ILF
         FC(Jstatus,I) = FC(Jstatus,IXF)
 7      CONTINUE
      DP = (PSTMAX - PSTMIN)/IMAX1
      FVCS = FC(Jstatus,IM) - FCMIN
      FVCD = FC(Jstatus,IM) - FC(Jstatus,1)
      SCALE = FVCS/FVCD
      E(1) = FCMIN
      DO 8 I = 2,IMF
         E(I) = E(I-1) + SCALE*(FC(Jstatus,I) - FC(Jstatus,I-1))
 8      CONTINUE
      DO 9 I = 1,IMF
         FC(Jstatus,I) = E(I)
 9      CONTINUE
      DO 10 I = 1,IMF
         VETIS(I) = (FC(Jstatus,I)*TLC + GRAMS)/AMV
 10   CONTINUE
C     JCOUNT=0
C     DO 15 I = 1,IMF
C        JCOUNT=JCOUNT+1
C        PR = PSTMIN + (I-1)*DPC
C        DENSC=GRAMS/(FC(Jstatus,I)*TLC+GRAMS)
C        WRITE(0,4) I, PR, VETIS(I), DENSC, FC(Jstatus,I)
C        IF(JCOUNT.LT.20) GO TO 15
C        PAUSE'PAUSE TO LOOK AT I, PR, VETIS(I), DENSC, FC(Jstatus,I)'
C        JCOUNT=0
 15   CONTINUE
C     PAUSE 'PAUSE TO LOOK AT I, PR, VETIS(I), DENSC, FC(Jstatus,I)'
      CALL INTERP(VETIS,IMF,DPC,IMAX,DP)
           JCOUNT=0
C     DO 20 I = 1,IMAX
C        JCOUNT=JCOUNT+1
C        PR = PSTMIN + (I-1)*DP
C        DENSC=GRAMS/(FC(Jstatus,I)*TLC+GRAMS)
C        WRITE(0,4) I, PR, VETIS(I), DENSC, FC(Jstatus,I)
C        IF(JCOUNT.LT.20) GO TO 20
C        PAUSE'PAUSE TO LOOK AT I, PR, VETIS(Jstatus,I), DENSC, FC(Jstatus,I)'
C        JCOUNT=0
C 20  CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
        SUBROUTINE INTERP(Y,IMAX1,DX1,IMAX2,DX2)
      DIMENSIONY(211), X1(211), X2(211), Y2(211)
      DO 10 I1 = 1,IMAX1
          X1(I1) = (I1 -1)*DX1
   10   CONTINUE
      DO 20 I2 = 1,IMAX2
          X2(I2) = (I2-1)*DX2
   20   CONTINUE
      DO 30 I2 = 1,IMAX2
```

Program REGPVCUR.FOR

```
          I1 = X2(I2)/DX1 + DX1/2.0
          I1 = I1 +1
          IF(I1 .EQ. 1) I1 = 2
          IF(I1 .EQ. IMAX1) I1 = IMAX1 - 1
          U = (X2(I2) - X1(I1))/DX1
          D1 = Y(I1+1) - Y(I1-1)
          D2 = Y(I1+1) - 2.0*Y(I1) + Y(I1-1)
          Y2(I2) = Y(I1) + 0.5*D1*U + 0.5*D2*U**2
   30   CONTINUE
      DO 40 I2 = 1,IMAX2
          Y(I2) = Y2(I2)
   40   CONTINUE
      RETURN
        END
C-----------------------------------------------------------------------
      SUBROUTINE SOLVEZ(A,B,C,D,Z0,Z2)
      Z2 = Z0
 10   F = A*Z2**3 + B*Z2**2 + C*Z2 + D
      AF = ABS(F)
        IF(AF .GE. 0.05) THEN
          FP = 2.0*A*Z2**2 + 2.0*B*Z2 + C
          Z2 = Z2 - F/FP
          GO TO 10
        END IF
      RETURN
      END
```

Appendix C - 3
Ventilation Model

PROGRAM SBBOLUSD

```
DIMENSION DV(7),CBOLUS(30),VMAX(10),CBOLreg(10)
      COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),Z0(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),V0(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAX0,SLOPE0,
     + JSTADYN,JQSTATIC,DPGRAV0(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
C  COMMON STATEMENTS FOR IODECLS
C
      INTEGER A1, A2
      INTEGER CRT, LUOUT, LUIN, LUPLT
      CHARACTER*1 BEEP
      CHARACTER*80 ITITLE
      CHARACTER*64 NAINP,NAOUT,NAPLT
      COMMON /DEVS/LUTRM, LUIN, LUOUT, LUPLT
      COMMON /CURSOR/ BEEP
      COMMON /LABELS/ITITLE,NAINP,NAOUT,NAPLT
C
  1   FORMAT(4F10.3)
  2   FORMAT(5F10.4,I10)
  3   FORMAT(13F6.3)
  4   FORMAT(I3,2F10.4)
  5   FORMAT(13F6.2)
  6   FORMAT(10E13.3)
  7   FORMAT(4I5,F10.2)
  9   FORMAT(/I10,3F15.3)
  10  FORMAT(4F25.5)
  14  FORMAT(2X,I10,7F14.5)
  15  FORMAT(2I10,8F12.4)
  18  FORMAT(2E15.3)
  20  FORMAT('  J', 9X,'Z(J)', 9X,' Q(J) ', 9X,' V(J) ', 9X,' H ',
     +9X,'  PVC ', 9X,'   FTV ', 9X,' P(J) ',8X,'DPRL(J)',7X,'RCF(J)')
  21  FORMAT(I3,F15.3,2F15.2,5F15.4,F13.4)
  23  FORMAT(1H1)
  25  FORMAT(/)
  26  FORMAT(10F8.1)
  28  FORMAT(10F8.4)
  29  FORMAT(1X,9F8.4)
  30  FORMAT(1X,I5,2F10.2)
  31  FORMAT(1X,I4,F8.1,4F8.2,4F8.1)
  32  FORMAT(2X,I5,5F9.2)
  33  FORMAT(I5,F6.1)
  34  FORMAT(5F11.5)
  35  FORMAT(10F8.1)
       CRT=0
C
C         DEFINE THE TYPE OF BREATH THAT WILL BE ANALYZED
C           JSTADYN             INHALATION  EXHALATION
C              1                 DYNAMIC    DYNAMIC
C              2                 STATIC     STATIC
C              3                 DYNAMIC     STATIC
C              4                 STATIC     DYNAMIC
C
```

Program SBBOLUSD.FOR

```
      JSTADYN=1
      DVdt=20          !Volume increment per DT time step
C
C  READ IN THE LUNG VOLUMES FOR THIS SUBJECT FROM THE DATA FILE CREATED
C  IN REGPFCUR.  ALPHA AND BETA ARE EXPANSION COEFFICIENTS AND
C  VMINLUNG IS THE MINIMUM  VOLUME OF THE LUNGS (GAS AND PARECHYMA),
C  AND WEIGHT IS THE TOTAL MASS OF THE LUNGS.
C
      CALL IODECLS(1,0,0)
      READ(LUIN,2) RV,FRC,TLC,DEADSP
      READ(LUIN,2) ALPHA,BETA,WEIGHT
      READ(LUIN,2) (DPGRAVO(I),I=4,6)
C
C    READ IN THE LUNG FUNCTIONAL DATA.IT IS ASSUMED THAT THE LUNG
C    TISSUE HAS NO HYSTERESIS AND ALL REGIONS HAVE THE SAME TISSUE
C    CHARACTERISTICS.  THE REGIONALP P-V CURVES HAVE BEEN CALCULATED
C    OUTSIDE THIS PROGRAM WITH THESE ASSUMPTIONS.
C
      READ(LUIN,4) ImaxFC,FCMIN,FCMAX
      DO 40 J=4,7
      DO 40 Jstatus = 1,2
   40 READ(LUIN,28) (Pst(Jstatus,J,I),I=1,ImaxFC)


      DFC=(FCMAX-FCMIN)/(ImaxFC-1)
      DFC2=2.0*DFC
      DFCS=DFC**2
      SLOPE=5.0/DFC
      DO 41 J=4,7
      DO 41 Jstatus=1,2
C ADD TWO LINEAR POINTS TO THE BEGINNING OF THE CURVE
      DO 39 I=ImaxFC,1,-1
 39   Pst(Jstatus,J,I+2)=Pst(Jstatus,J,I)
      Pst(Jstatus,J,2)=Pst(Jstatus,J,3)-SLOPE*DFC
      Pst(Jstatus,J,1)=Pst(Jstatus,J,3)-2.0*SLOPE*DFC
C ADD TWO LINEAR POINTS ONTO THE END OF THE CURVE

      Pst(Jstatus,J,ImaxFC+2)=Pst(Jstatus,J,ImaxFC+1)+SLOPE*DFC
      Pst(Jstatus,J,ImaxFC+3)=Pst(Jstatus,J,ImaxFC+1)+2*SLOPE*DFC
   41 Pst(Jstatus,J,ImaxFC+4)=Pst(Jstatus,J,ImaxFC+1)+3*SLOPE*DFC

      ImaxFC=ImaxFC+4
      ImaxFCM1=ImaxFC-1
      ImaxFCP1=ImaxFC+1

      FCMIN=FCMIN-2.0*DFC
      FCMAX=FCMAX+2.0*DFC


C
C          READ IN THE EXHALATION PARENCHYMAL P-FV RELATION
C                    FOR CALCULATING AIRWAY DIAM

      READ(LUIN,4) Jmax,PTMMIN,PTMMAX
      READ(LUIN,28)(FVt(J),J=1,Jmax)

C   ADD HYSTERESIS TO MAKE P-FV CURVE AN INHALATION CURVE
C         HYSTERESIS IN A VC BREATH IS 4.0cm H2O

      DPHYS=4.0
      PTMMIN=PTMMIN+DPHYS
      PTMMAX=PTMMAX+DPHYS
      .
C     WRITE(CRT,4) Jmax,PTMMIN,PTMMAX
C     WRITE(CRT,28)(FVt(J),J=1,Jmax)
C     PAUSE'PAUSE TO LOOK AT IMFVA,PTMMIN,PTMMAX AND FVt(I),I=1,IMFVA'
```

Program SBBOLUSD.FOR

```
      JmaxM1=Jmax-1

C  DEADSPACE=160cc     TOTAL ANATOMICAL DEADSPACE
C  RDS=25cc                ANATOMICAL DEADSPACE WITHIN A REGION THAT REPRESENTS
C                     A QUARTER OF THE TOTAL LUNG MASS
C  UAWDS=60cc              ANATOMICAL DEADSPACE FOR THE UPPER AIRWAYS
C                     (MOUTH,THROAT,TRACHEA)

      DEADSPACE=160.0
      RDS=25.0
      UAWDS=60.0

      TLCT=TLC
      FRCT=FRC
      TLC=TLC-DEADSP
      FRC=FRC-DEADSP
      RV=RV-DEADSP
      FCFRC=FRC/TLC
      VMINLUNG=FCMIN*TLC+WEIGHT
C
C  OPEN OUTPUT FILE
C
       CALL IODECLS(0,1,0)
C
C   INPUT DATA FOR WEIBEL'S LUNG MODEL A FROM WEIBEL.DAT
C
      NI5=5
C   OPEN INPUT FILE 'WEIBEL.DAT'
      OPEN(NI5,FILE='WEIBEL.DAT')
      NGEN=23                        !Number of airway generations
      READ(NI5,1) VOM                !Total gas volume
      READ(NI5,3) (WL(N),N=1,23)      !Airway lengths in cms
      READ(NI5,3) (WD(N),N=1,23)      !Airway diameters in cms
      WTRDIAM = 1.8                   !Trachea diameter in cms

C   READ IN THE BOLUS CONCENTRATION AT THE END OF THE TRACHEA
C   THIS PROFILE WAS CALCULATED OUTSIDE THIS PROGRAM

       NI4=4
       OPEN(NI4,FILE='CBOLUS.DAT')
       READ(NI4,33) JmaxB,DVbolus
       READ(NI4,34) (CBOLUS(JB),JB=1,JmaxB)
       JmaxBP1=JmaxB+1
C     WRITE(CRT,33) JmaxB,DVbolus
C     WRITE(CRT,34) (CBOLUS(JB),JB=1,JmaxB)
C     PAUSE'LOOK AT INPUT DATA'

C
C  SUBSCRIBTS THAT DEFINE THE DIFFERENT AREAS OF THE LUNG MODEL
C         SUBSCRIBT 1 - ENTIRE LUNG
C         SUBSCRIBT 2 - UNIT COMPOSED OF UPPER TWO REGIONS 1 and 2
C         SUBSCRIBT 3 - UNIT COMPOSED OF LOWER TWO REGIONS 3 and 4
C         SUBSCRIBT 4 - REGION 1
C         SUBSCRIBT 5 - REGION 2
C         SUBSCRIBT 6 - REGION 3
C         SUBSCRIBT 7 - REGION 4
C
      F(1)=1.0            !F(I) IS THE LUNG MASS OF EACH AREA
      F(2)=.5
      F(3)=.5
      F(4)=.25
      F(5)=.25
      F(6)=.25
      F(7)=.25
C
```

Program SBBOLUSD.FOR

```
C     INITIALIZE VARIABLES
C
      DO 36 J=1,7
      AMOUNT(J)=0.0
      RTLC(J)=F(J)*TLC          !Regional TLC, does not include the upper
      RWEIGHT(J)=F(J)*WEIGHT
      RMINVOL(J)=F(J)*VMINLUNG
      RCV(J)=.5
 36   DPRL(J)=.1

      DPFG=0.0
      FDG=1.0
      IBOLUS=1
      C=0.0

C
C  CHECK INPUT DATA
C
C     WRITE(CRT,10) FRC,TLC
C     PAUSE'PAUSE TO LOOK AT FRC,TLC  '
C     WRITE(CRT,28) ( F(J),J=1,7)
C     PAUSE'PAUSE TO LOOK AT F(J) '
C     WRITE(CRT,26) ( RTLC(J),J=1,7)
C     PAUSE'PAUSE TO LOOK AT RTLC(J) '
C     WRITE(CRT,3) (WL(N),N=1,23)
C     PAUSE'PAUSE TO LOOK AT  WL'
C     WRITE(CRT,3) (WD(N),N=1,23)
C     PAUSE'PAUSE TO LOOK AT  WD'
      DO 44 J=4,7
      DO 44 Jstatus=1,2
      WRITE(CRT,29) (Pst(Jstatus,J,I),I=1,ImaxFC)
      PAUSE'PAUSE TO LOOK AT PST '
 44   CONTINUE
C     WRITE(CRT,29) (FVt(J),J=1,Jmax)
C     PAUSE'PAUSE TO LOOK AT THE FVA'
C
C     READ IN THE EXPERIMENTAL CONDITIONS

      DENSITY =0.001121               !gms/cm**3
      VISCSTY =0.0001914              !gm/(sec*cm)
      VISK=VISCSTY/DENSITY            !Kinematic VIScosity (cm**2/sec)
      GI=1/980.2                      !dynes/cm**2=(1/980.2)cmH20
      DD=GI*DENSITY/2
      DK=DENSITY*0.85               !0.85 experimental constant for conv acc
      EXC=1.85                      !1.85 experimental constant for vis losses
      COEFA=EXC*5.659*VISCSTY       !COEFficient for Vis Losses
      NGENDP=16                     !# of gen used in DP calc
      NGENDP1=NGENDP+1
C
C     WRITE(CRT,6) VISCSTY,DENSITY,VISK
C     PAUSE'PAUSE TO LOOK AT VISCSTY, DENSITY, VISK'
C
      CALL FLOWW(KMAX,KEINS,VSTART)
      V(1)=VSTART-DEADSP        !gas Volume of area 1 at the start
C
C  USE CONSTANT VOLUME INCREMENTS OF DVdt SIZE IF THIS IS A STATIC BREATH
C
      IF(JSTADYN.NE.2) GOTO 49
      VOLIN(1)=0.0
      DO 47 K=2,KEINS
 47   VOLIN(K)=VOLIN(K-1)+DVdt
      KEINSP1=KEINS+1
      DO 48 K=KEINSP1,KMAX+1
 48   VOLIN(K)=VOLIN(K-1)-DVdt
 49   CONTINUE
```

Program SBBOLUSD.FOR

```
C
C     WRITE(CRT,2) TIDVOL, DT, TMAX
C      PAUSE'PAUSE TO LOOK AT TIDVOL, V(1), DT, TMAX '
C
C  CALCULATE THE MAXIMUM LENGTH AND DIAMETER OF EACH GENERATION
C
      SCALE=(TLCT/VOM)**(1./3.)     !The linear scaling of VOM to TLCT
      TRCDMAX=WTRDIAM*SCALE         !Maximum trachea diameter
      DO 50 N=1,NGEN
      XLMAX(N)=SCALE*WL(N)          !Maximum airway lengths
   50 DMAX(N)=SCALE*WD(N)           !Maximum airway diameters

C     WRITE(CRT,3) (XLMAX(N),N=1,NGEN)
C      PAUSE'PAUSE TO LOOK AT (XLMAX(N),N=1,NGEN)'
C     WRITE(CRT,3) (DMAX(N),N=1,NGEN)
C      PAUSE'PAUSE TO LOOK AT (DMAX(N),N=1,NGEN)'
C
C          CALCULATION OF OMEGA - XKM=2.0 AT P=5CMH2O
      I=1
   54 IF(Pst(Jstatus,6,I).GE.5.0) GO TO 56
      I=I+1
      GO TO 54
   56 OMEGA=(Pst(Jstatus,6,I+1)-Pst(Jstatus,6,I-1))/(2.*DFC)
C     WRITE(CRT,1) OMEGA        !NOTE OMEGA=19.0 cm H2O CHECK!!
C     PAUSE'PAUSE TO LOOK AT OMEGA'
C
      XKB=2.474          !2.474 cm H2O (3.21 and 3.22)
C
C   CALCULATION OF D/DMAX-PTM RELATION FOR POSITIVE PTM.  THE
C   AIRWAYS ARE ASSUMED TO EXPAND UNIFORMLY WITH THE PARENCHYMA
C   AND THE TRANSMURAL PRESSURE IS EQUAL TO THE TRANSPULMONARY PRESSURE.

      DPptm=(PTMMAX-PTMMIN)/(JMAX-1)
      PTMlow=PTMMIN-DPHYS          !Use the exhalation P-FVt curve

C  DETERMINE AT WHAT J VALUE THE PTM BECOME POSITIVE
      J=0
   58 J=J+1
      Pptm = PTMlow+(J-1)*DPptm
      IF(Pptm.LT.0.0) GOTO 58
      Jstart=J-1
      IF(Jstart.EQ.0) Jstart=1
      JmaxPTM=JMAX-(Jstart-1)

      JJ=0
      DO 60 J=Jstart,Jmax
      JJ=JJ+1
   60 FDMAX(JJ)=FVt(J)**(1./3.)
      WRITE(CRT,29) (FDMAX(J),J=1,JmaxPTM)
      PAUSE'PAUSE TO LOOK AT FDMAX(J),J=1,JmaxPTM'

C  CALCULATE PARAMETERS NEEDED FOR THE TUBE LAW (NEGATIVE PTM's)
      FDMAX0=FDMAX(1)
      SLOPE0=DPptm/(FDMAX(2)-FDMAX(1))

      CHECKP=.001          !Pressure error in iteration, cmH2O
      DPHYS=0.00           !Hysteresis for a VC breath
C
C    CALCULATE THE REGIONAL VOLUMES, TRANSMURAL PRESSURES
C    AND AIRWAY DIAMETERS BEFORE THE BREATH STARTS.
C
      VELUNG=(V(1)+WEIGHT)/VMINLUNG
      FCLUNG=V(1)/TLC
C     WRITE(CRT,29) V(1),WEIGHT,VMINLUNG,TLC
C     PAUSE'PAUSE TO LOOK AT V(1),WEIGHT,VMINLUNG,TLC'
```

```
      DO 63 J=4,6
   63 DPGRAV(J)=DPGRAVO(J)/VELUNG**(ALPHA+BETA)
C     WRITE(CRT,6)V(1),WEIGHT,VMINLUNG,FCLUNG,(DPGRAV(J),J=4,6)
C     PAUSE' PAUSE; V(1),WEIGHT,VMINLUNG,FCLUNG,DPGRAV(J),J=4,6 '

      K=1    !K=1 IS THE LUNG VOLUME AT STRART OF BREATH, TIME=0
      T=0.0                      !TIME ZERO IS THE START OF THE BREATH
C
      Jstatus=1                        !Inhalation
      Iprint=1
      CALL STATVOL(FClung,Jstatus)
      CALL TRANMPR(OMEGA,DPFG)
      CALL AIRWAY(TRCDMAX,TRCDIAM)
C
C-----------------------------------------------------------------
C   CALCULATIONS OF THE REGIONAL FLOWS AT EACH DT TIME INCREMENT
C          INITIALIZE PARAMETERS
C
      JCYCLE=1            !JCYCLE=1 IS INHALATION, =-1 IS EXHALATION
      JINSIGN=1
      JEXSIGN=-1
C  Z DEFINES THE AMOUNT OF FLOW GOING TO EACH OF THE AREAS OF THE MODEL
C
      Z(1)=1.0       !Z defines the fraction of flow going to
      Z(2)=0.5       !  a region, Z(Region)
      Z(3)=0.5
      Z(4)=0.25
      Z(5)=.25
      Z(6)=.25
      Z(7)=.25
C      PAUSE 'PAUSE AFTER THE Zs ARE DEFINED'
C
C     AT T = 0 THE SYSTEM IS IN ELASTIC EQUILIBRIUM.
C     THE DIFFERENCES IN QU AND QL IN EACH SET IS DUE TO
C     THE DIFFERENT AIRWAY RESISTANCES
C
      K=1     !K=1 IS THE INITIAL TIME STEP (TIME=0)
      T=0.0              !TIME ZERO IS THE START OF THE BREATH

      WRITE(LUOUT,31)K,V(1),(P(J),J=4,7),(V(J),J=4,7)
      WRITE(LUOUT,2499)
C-----------------------------------------------------------------
 1000 T=T+DT
      K=K+1
      IF(K.GT.KMAX) GOTO 5000
      DO 1002 J=1,7
 1002 V0(J)= V(J)
      IF(K.LT.257) GOTO 1003
      KKK=1
 1003 CONTINUE
      Q(1)=FLOW(K)       ! Q(1) IS FLOW IN THE TRACHEA-TOTAL FLOW
      AQ=ABS(Q(1))
      DELVOL=VOLIN(K)-VOLIN(K-1)
      V(1)=V(1)+DELVOL
            WRITE(CRT,32) K,Q(1),DELVOL,V(1)

C        PAUSE'PAUSE TO SEE K,Q(1),DELVOL AND V(1) FOR THE NEXT TIME STEP'
C
C  CALCULATE THE VOLUME EXPANSION OF THE LUNG AND THE FOUR REGIONS
C  AND THE GRAVITATIONAL EFFECT ON THE REGIONAL PARENCHYMAL PRESSURES
C
      VE=(V(1)+WEIGHT)/VMINLUNG
      DO 1005 R=4,6
      VEXP=(RVE(R)+RVE(R+1))/2.0
 1005 DPGRAV(R)=DPGRAVO(R)/VEXP**(ALPHA+BETA)
```

```
C        WRITE(CRT,29) (DPGRAV(R),R=4,6)
C        PAUSE'PAUSE TO LOOK AT THE DPRGAV(R),R=4,6'
      IF (K.EQ.KEINS) GO TO 3000
 1006 CONTINUE      ! return after setting exhalation values
      IF(JSTADYN.EQ.2) GO TO 1010
C        WRITE(CRT,14) K,Q(1),DELVOL
C        PAUSE'PAUSE TO LOOK AT K, Q(1), DELVOL'
      IF(AQ.LT.50.0) GO TO 1010
      GO TO 1040
 1010 JQSTATIC=2
C
C     JQSTATIC=2 INDICATES QUASISTATIC FLOW,FLOWS < 50 cc/sec.
C          USE STATVOL TO CALCULATE THE FLOW RATES
C
      DELVOL=ABS(DELVOL)
      FCLUNG=V(1)/TLC
      CALL STATVOL(FClung,Jstatus)
      DO 1020 J=2,7
      DPRL(J)=0.0
 1020 RCV(J)=0.0
      GO TO 1405
 1040 CONTINUE
      JQSTATIC=1
C
C     JQSTATIC=1 INDICATES DYNAMIC FLOW (FLOWS > 50 cc/sec).
C
C   TO DETERMINE THE REGIONAL FLOWS ITERATE ABOUT THE FLOW TO
C   AREA 2 (Q(2)).  NEED TWO GUESSES FOR Q(2) TO GET THE ITERATION
C   STARTED.  THE FIRST GUESS Q1 IS BASED ON THE FLOWS AT THE
C   PREVIOUS TIME STEP. Q2 AND Q3 ARE THE LATEST GUESSES FOR Q(2).
C   THE ITERATION FUNCTION F IS THE ERROR WHEN THE PRESSURES IN EQ.
C   3.32 ARE NOT PROPERLY BALANCED.  ONCE THE ERROR IS BELOW CHECKP
C   THE ITERATION IS STOPPED.
C
      Q2=Z(2)*Q(1)          ! first guess for Q(2)
      QG(2)=Q2                ! define QG(2) and QG(3) for use in
      QG(3)=Q(1)-QG(2)        !     REGVOL
C       WRITE(CRT,29) Z(2),Q(1),Q2
C       PAUSE'PAUSE TO LOOK AT Z(2),Q(1),Q2 IN MAIN'
C
      A1=4      !beginning Area for PRESSLS calculations
      A2=7      !end Area for PRESSLS calculations
      LSIT=1            !LSITuation =1, 1st time enter PRESSLS in K time step
      CALL PRESSLS(A1,A2,LSIT,Q2)
      CALL REGFLOW(CHECKP)
C
C           LSIT =2 FOR ALL OTHER ENTRIES INTO PRESSLS
C
C     DPRL IS THE PRESSURE DROP FROM THE ALVEOLI IN THE ITH
C     POSITION OF THE TRACHEA
C     THE FUNCTION USED IN THE ITERATION IS THE ERROR WHEN THE
C     PRESSURES BETWEEN AREAS 2 AND 4 ARE NOT BALANCED.
C
      IGUESS=1
      DPRL(5)=DPR(5)+DPG(2,2)
      DPRL(7)=DPR(7)+DPG(3,2)
      G=P(5)-P(7)-(DPGRAV(5)+DPGRAV(6))
      F2=G+(DPRL(7)-DPRL(5))
      AF2=ABS(F2)
      IF(K.LT.71) GOTO 1290
C     WRITE(LUOUT,7) K,LSIT
C     WRITE(LUOUT,32) IGUESS,F2,DPRL(5),DPRL(7),DPG(2,2),DPG(3,2)
 1290 CONTINUE
C        WRITE(CRT,29) G,P(5),P(7),DPGRAV(5),DPGRAV(6),DPRL(5),DPRL(7)
C        PAUSE'PAUSE TO LOOK AT G,P(5),P(7),DPGRAV(5),DPGRAV(6),DPRL(5)
```

```
C     +    ,DPRL(7), READY TO GET SECOND GUESS FOR Q(2) IN MAIN'
      Q3=Q2               !Define Q3 in case AF2.LT.CHECKP
      IF(AF2.LT.CHECKP) GO TO 1400
C                                   SECOND GUESS
      IF(F2.GT.0.0) Q3=1.02*Q2
      IF(F2.LT.0.0) Q3=0.98*Q2
C
 1300 F1=F2
      Q1=Q2
      Q2=Q3
      IGUESS=IGUESS+1
C
      QG(2)=Q2  !define QG(2) and QG(3) for use in REGVOL calc
      QG(3)=Q(1)-QG(2)
C        WRITE(CRT,29) Q2,QG(3)
C        PAUSE'PAUSE TO LOOK AT THE NEXT GUESS FOR Q2 AND Q3 IN MAIN'
      LSIT=2
      A1=4
      A2=7
      CALL PRESSLS(A1,A2,LSIT,Q2)
      CALL REGFLOW(CHECKP)
      DPRL(5)=DPR(5)+DPG(2,2)
      DPRL(7)=DPR(7)+DPG(3,2)
      F2=G+(DPRL(7)-DPRL(5))
      AF2=ABS(F2)
C        WRITE(CRT,15) IJUMP
C        WRITE(CRT,29) Q1,Q2,F1,F2,DPRL(5),DPRL(7)
C        PAUSE'PAUSE TO LOOK AT LCOUNT,Q1,Q2,F1,F2 IN MAIN'
      IF(AF2.LT.CHECKP) GO TO 1400
C                                   NEXT GUESS
      Q3=(Q1*F2-Q2*F1)/(F2-F1)
      IF(K.LT.71) GOTO 1390
C     WRITE(LUOUT,7) K,LSIT
C     WRITE(LUOUT,32) IGUESS,F2,DPRL(5),DPRL(7),DPG(2,2),DPG(3,2)
 1390 CONTINUE
      IF(IGUESS.GT.25) GO TO 5000
      GO TO 1300
 1400  CONTINUE
C
C       ITERATION FOR Kth TIME STEP COMPLETE
C
      Q(2)=Q3
      Q(3)=Q(1)-Q(2)
      DO 1402 J=1,7
 1402 ZO(J)=Z(J)

      GOTO 1420

 1405 CONTINUE

C            CALCULATE THE Zs FOR THIS TIME STEP

C                     QUASISTATIC CASE

      DO 1407 J=4,7
      DV(J)=V(J)-VO(J)
 1407 Z(J)=DV(J)/DELVOL   !CHANGE MADE CHECK OUT!!!!!!!!!!!!!!!!

      SUM=Z(4)+Z(5)+Z(6)+Z(7)

      DO 1408 J=4,7
 1408 Z(J)=Z(J)/SUM
      IF(JSTADYN.EQ.2) GOTO 2100

      DO 1410 J=4,7
```

```
 1410 Q(J)=DV(J)/DT
      Q(2)=Q(4)+Q(5)
      Q(3)=Q(6)+Q(7)
      GOTO 2100

 1420 CONTINUE
C                            DYNAMIC CASE
      DO 1450 J=1,7
 1450 Z(J)=Q(J)/FLOW(K)
C        WRITE(CRT,14) K
C        WRITE(CRT,29) (Z(J),J=1,7)
C        PAUSE'PAUSE TO LOOK AT THE Zs AT THE END OF THE Kth TIME STEP'
      IF(JSTADYN.EQ.1.AND.JQSTATIC.EQ.1) GO TO 1800
      IF(Q(1).EQ.0.0) GO TO 1800
      SUM=0.0
      DO 1560 J=4,7
 1560 SUM=SUM+Z(J)
      DO 1600 J=4,7
 1600 Z(J)=Z(J)/SUM
 1800 CONTINUE
 1810 VOLD(1)=V(1)
      VOLD(2)=V(2)
      VOLD(3)=V(3)
C
C     INTEGRATE THE FLOW TO EACH REGION (AREA=4,7) AND CALCULATE THE
C     NEW RECOIL PRESSURE OF THE PARENCHYMA OF THAT REGION
C
      DO 2000 J=4,7
      VOLD(J)=V(J)
      FVREG=V(J)/RTLC(J)
      IF(JQSTATIC.EQ.2) GO TO 1980
      IF(JSTADYN.EQ.2) GO TO 1980
      DV(J)=DT/2.0*(ZO(J)*FLOW(K-1)+Z(J)*FLOW(K))
      V(J)=V(J)+DV(J)

      I=0
 1820 I=I+1
      IF(I.EQ.ImaxFC) GOTO1840
      FCI=FCMIN+DFC*(I-1)
      IF(FCI.LT.FVREG) GOTO 1820
      IF(I.EQ.1) GO TO 1830

      UU=FVREG-(FCI-DFC)
      P(J)=Pst(Jstatus,J,I-1)
     *         + UU*(Pst(Jstatus,J,I)-Pst(Jstatus,J,I-1))/DFC
      GO TO 1980

 1830 UU=FVREG-FCMIN
      P(J)=Pst(Jstatus,J,1)
     *          + UU*(Pst(Jstatus,J,2)-Pst(Jstatus,J,1))/DFC
      GO TO 1980

 1840 UU=FVREG-FCMAX
      P(J)=Pst(Jstatus,J,ImaxFC)
     *          + UU*(Pst(Jstatus,J,ImaxFC)-Pst(Jstatus,J,ImaxFCM1))/DFC

 1980 CONTINUE
      PVC=(V(J)-VO(J))/VO(J)
      FTV=(V(J)-VO(J))/VOLIN(K)
C        WRITE(CRT,21) J,Q(J),V(J),P(J),DPRL(J)
C        PAUSE'PAUSE TO LOOK AT J,Q(J),V(J),P(J),DPRL(J)'
 2000 CONTINUE
C
C   CALCULATE THE NEW VOLUMES OF AREAS 2 AND 3
C
```

```
      V(2)=V(4)+V(5)
      V(3)=V(6)+V(7)
C
C  CALCULATE THE VELOCITY AND DPs IN THE TRACHEA AND THE 1st GEN
C
      VTRACH=ABS(Q(1)/(.785*TRCDIAM**2))
      COEF=-JCYCLE*0.0107
      DPFR=COEF*U(1,1)*VISCSTY/AWD(1,1)*(U(1,1)*AWL(1,1)/VISK)**0.5
      DPBE=DD*(FDG*VTRACH**2-1.7*U(1,1)**2)
      DPFG=DPFR+DPBE
C
C   CALCULATE THE TRANSMURAL PRESSURES AND THE NEW AIRWAY DIAMETERS
C     AND LENGTHS THAT WILL BE USED IN THE NEXT TIME STEP K+1
C
      CALL TRANMPR(OMEGA,DPFG)
C        PAUSE'PAUSE BEFORE YOU ENTER AIRWAY'
      CALL AIRWAY(TRCDMAX,TRCDIAM)
C
 2100 CONTINUE
      IF(JCYCLE.EQ.-1) GOTO 2200
      IF(IBOLUS.EQ.0) GOTO 2300

C    CALCULATE THE AMOUNT OF INDICATOR MATERIAL THAT ENTERS EACH REGION

       CO=C
       JB=1
 2110  JB=JB+1
       IF(JB.EQ.JmaxBP1) GOTO 2120
       VOLjb=(JB-1)*DVbolus
       IF(VOLIN(K).GT.VOLjb) GOTO 2110
       UU=(VOLIN(K)-VOLJB)/DVbolus
       C=CBOLUS(JB-1)+UU*(CBOLUS(JB)-CBOLUS(JB-1))
       GOTO 2130
 2120  IBOLUS=0
       C=0

 2130  CONTINUE
       DO 2140 J=4,7
 2140  AMOUNT(J)=AMOUNT(J)+DV(J)*(C+CO)/2.0
       GOTO 2300

C   CALCULATE THE GAS CONCENTRATION AT THE MOUTH DURING
C   EXHALATION (CONMO) - EACH REGION HAS RDS OF ANATOMICAL DEADSPACE
C   AND THE UPPER AIRWAYS HAS A DEADSPACE UAWDS

 2200  CONMO=0.0
       DO 2150 J=4,7
       FK=1.0
       RVEXH=VMAX(J)-V(J)
       IF(RVEXH.LT.RDS) FK=0.0
 2150  CONMO=CONMO+FK*Z(J)*CbolREG(J)
       EXHVOL=VMAX(1)-V(1)+UAWDS
       WRITE(LUOUT,30) K,EXHVOL,CONMO
       WRITE(CRT,30) K,EXHVOL,CONMO

 2300  CONTINUE

      Iprint=Iprint+1
      WRITE(LUOUT,31)K,V(1),(P(J),J=4,7),(V(J),J=4,7)
      WRITE(LUOUT,31)K,Q(1),(Z(J),J=4,7),(DPR(J),J=4,7)
      IF(IPRINT.LT.1000) GOTO 2900
      Iprint=0
      DELGRAVITY=DPGRAV(5)+DPGRAV(6)

      IF(JSTADYN.EQ.2) GOTO 2500
```

```
C     CALCULATE PULMONARY RESISTANCE

      DO 2400 J=4,7
 2400 RES(J)=DPR(J)/(Q(J)*.001)

      RESI = DPG(2,2)/(Q(2)*.001) + (RES(4)*RES(5))/(RES(4)+RES(5))
      RESII= DPG(3,2)/(Q(3)*.001) + (RES(6)*RES(7))/(RES(6)+RES(7))

      RESPUL=DPFG/(Q(1)*.001) + (RESI*RESII)/(RESI+RESII)
 2499 FORMAT(/)
 2500 CONTINUE
      WRITE(LUOUT,2499)
      WRITE(LUOUT,2499)
      WRITE(LUOUT,29) DELGRAVITY,DPGRAV(5),DPGRAV(6)
      WRITE(LUOUT,29) DPR(4),DPR(5),DPR(6),DPR(7),DPRL(5),DPRL(7)
      WRITE(LUOUT,2499)
C     WRITE(LUOUT,29) (XXM1(J),J=4,7)
C     WRITE(LUOUT,29) (XK(4,N),N=2,NGENDP)
      WRITE(LUOUT,28) PTM(1,1),PTM(2,2),(PTM(4,N),N=3,NGENDP)
      WRITE(LUOUT,28) AWD(1,1),AWD(2,2),(AWD(4,N),N=3,NGENDP)
      WRITE(LUOUT,28) PTM(1,1),PTM(3,2),(PTM(7,N),N=3,NGENDP)
      WRITE(LUOUT,28) AWD(1,1),AWD(3,2),(AWD(7,N),N=3,NGENDP)
      WRITE(LUOUT,2499)
      WRITE(LUOUT,35)  VTRACH,U(1,1),U(2,2),U(3,2)
      WRITE(LUOUT,35) (U(4,N),N=3,NGENDP)
      WRITE(LUOUT,35) (U(7,N),N=3,NGENDP)
      WRITE(LUOUT,2499)
      WRITE(LUOUT,28) DPFR,DPV(2,2),(DPV(4,N),N=3,NGENDP)
      WRITE(LUOUT,28) DPBE,DPC(2,2),(DPC(4,N),N=3,NGENDP)
      WRITE(LUOUT,28) DPFG,DPG(2,2),(DPG(4,N),N=3,NGENDP)
      WRITE(LUOUT,2499)
      WRITE(LUOUT,28) DPFR,DPV(3,2),(DPV(7,N),N=3,NGENDP)
      WRITE(LUOUT,28) DPBE,DPV(3,2),(DPC(7,N),N=3,NGENDP)
      WRITE(LUOUT,28) DPFG,DPG(3,2),(DPG(7,N),N=3,NGENDP)
      WRITE(LUOUT,2499)
      WRITE(LUOUT,28) (RES(J),J=4,7),RESI,RESII,RESPUL
 2900 WRITE(LUOUT,2499)
      GO TO 1000          !goto next time step K+1
 3000 CONTINUE
C
C      START OF EXHALATION; SET THE EXHALATION PARAMETERS
C
      JCYCLE=-1           !Exhalation coefficient
      FDG=1.7
      Jstatus=2           !Exhalation indicator

C   MAKE THE INHALATION P-FV CURVE AN INHALATION CURVE

       PTMMIN=PTMMIN-DPHYS
       PTMMAX=PTMMAX-DPHYS

C   SAVE THE MAXIMUM REGIONAL VOLUMES

      VMAX(1)=V(1)
      DO 3100 J=4,7
 3100 VMAX(J)=V(J)

C   CALCULATE THE END INHALATION TRACER CONCENTRATION IN EACH REGION

C      NORMALIZE THE AMOUNT TO 1.0 microcurie

       SUM=0.0
       DO 3120 J=4,7
 3120  SUM=SUM+AMOUNT(J)
       DO 3130 J=4,7
```

Program SBBOLUSD.FOR

```
 3130  AMOUNT(J)=AMOUNT(J)/SUM

      DO 3150 J=4,7
      CBOLreg(J)=AMOUNT(J)/(V(J)-RDS)
      CBOLreg(J)=1000000 * CBOLreg(J)
      WRITE(LUOUT,14) J,V(J),CBOLreg(J),AMOUNT(J)
 3150 WRITE(CRT,14) J,V(J),CBOLreg(J),AMOUNT(J)
      PAUSE'PAUSE TO LOOK AT J,V(J),CBOLreg(J),AMOUNT(J)'

      GOTO 1006
 4500 PAUSE'ITERATION NOT CONVERGING!! IGUESS=25'

 5000 CONTINUE
        STOP
      END
C-------------------------------------------------------------------
      SUBROUTINE INTERP(Y,IMAX1,DX1,IMAX2,DX2)
      DIMENSIONY(211), X1(211), X2(601), Y2(601)
      DO 10 I1 = 1,IMAX1
           X1(I1) = (I1 -1)*DX1
   10   CONTINUE
      DO 20 I2 = 1,IMAX2
           X2(I2) = (I2-1)*DX2
   20   CONTINUE
      DO 30 I2 = 1,IMAX2
           I1 = X2(I2)/DX1 + DX1/2.0
           I1 = I1 +1
           IF(I1 .EQ. 1) I1 = 2
           IF(I1 .EQ. IMAX1) I1 = IMAX1 - 1
           U = (X2(I2) - X1(I1))/DX1
           D1 = Y(I1+1) - Y(I1-1)
           D2 = Y(I1+1) - 2.0*Y(I1) + Y(I1-1)
           Y2(I2) = Y(I1) + 0.5*D1*U + 0.5*D2*U**2
   30   CONTINUE
      DO 40 I2 = 1,IMAX2
           Y(I2) = Y2(I2)
   40   CONTINUE
      RETURN
        END
C-------------------------------------------------------------------
      SUBROUTINE FLOWW(KMAX,KEINS,VSTART)
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),Z0(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),V0(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAX0,SLOPE0,
     + JSTADYN,JQSTATIC,DPGRAV0(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
       INTEGER CRT
       CRT=0
  1    FORMAT(I3,F8.2,F8.3,F8.2)
  2    FORMAT(9F8.1)
  3    FORMAT(3X,I5)
C
C    READ THE FLOW DATA FROM THE DATA FILE CALLED FLOW.DAT
C          Jmax = # OF FLOW DATA POINTS FROM EXPERIMENTAL RECORD
C          VSTART=GAS VOLUME OF THE LUNG AT START OF BREATH
C          DTC = TIME INCREMENT FOR MEASURED FLOW DATA
C          TIDVOL = MEASURED TIDAL VOLUME
```

Program SBBOLUSD.FOR

```
C
      READ(4,1) JmaxQ,VSTART,DTC,TIDVOL
      READ(4,2) (FLOW(I),I=1,JmaxQ)
C      WRITE(CRT,1) JmaxQ,VSTART,DTC,TIDVOL
C      WRITE(CRT,2) (FLOW(I),I=1,JmaxQ)
C      PAUSE'PAUSE TO LOOK AT Jmax,VSTART,DTC,TIDVOL  AND FLOWS IN SUB FLOWW'
C
C    DETERMINE WHERE THE MEASURED DATA BECOMES NEGATIVE, JEINS.
C
      J=0
 100  J=J+1
      IF(FLOW(J).GE.0.0) GOTO 100
      JEINS=J-1
C
C    DO AN INITIAL SCALING TO MATCH THE FLOWS TO THE TIDAL VOLUME
C
      VSUM=0.0
      DO 110 J=2,JEINS
 110  VSUM=VSUM+DTC/2*(FLOW(J-1)+FLOW(J))
      SCALE = TIDVOL/VSUM
      DO 120 J=1,JmaxQ
 120  FLOW(J)=SCALE*FLOW(J)
C
C   INTERPRET FLOW DATA TO A FINER KMAX GRID.  THE TIME STEP DT FOR THIS
C   US BASED ON AN AVERAGE VOLUME INCREMENT OF DELVOL OVER INHALATION.
C
      JmaxQ1=JmaxQ-1
      TMAX=DTC*JmaxQ1
      TIMEINH = (JEINS-1)*DTC
      DT=TIMEINH*(DVdt/TIDVOL)
      KMAX1=TMAX/DT
      DT=TMAX/KMAX1
      KMAX=KMAX1+1
      DTH=DT/2.0

      CALL INTERP(FLOW,JmaxQ,DTC,KMAX,DT)

C     WRITE(CRT,2) (FLOW(I),I=1,KMAX)
C     WRITE(CRT,2) DTC,DT
C     PAUSE'PAUSE TO LOOK AT FLOWS AND DTC AND DT IN SUB FLOWW'
C
C  DETERMINE THE K WHERE THE FLOW FIRST BECOMES NEGATIVE,
C  THE END OF INHALATION (KEINS). LOOK FOR KCHECK FLOWS LESS THAN ZERO.
C
      KCHECK=3
      KC=0
      K=0
 150  K=K+1
C      WRITE(CRT,3)K
      IF(FLOW(K).GT.0) GOTO 200
      KC=KC+1
      IF(KC.EQ.KCHECK) GOTO 250
      GOTO 150
 200  KC=0
      GOTO 150
 250  KEINS=K-(KCHECK-1)
C      PAUSE'PAUSE TO LOOK AT ALL THE Ks'
C
C  INTEGRATE FLOW TO GET TIDAL VOLUME BASED ON FLOW DATA (FTIDVOL)
C
      KEINS1=KEINS-1
      FTIDVOL=0.0
      DO 300 K=1,KEINS1
 300  FTIDVOL=FTIDVOL+DTH*(FLOW(K)+FLOW(K+1))
```

Program SBBOLUSD.FOR

```
C
C   NORMALIZE FLOWS WITH RESPECT TO THE MEASURED TIDAL VOLUME
C
      SCALE=TIDVOL/FTIDVOL
      DO 400 K=1,KMAX
 400  FLOW(K)=SCALE*FLOW(K)
C
C  CALCULATE THE INHALED VOLUME (VOLIN) BASED ON THE SCALED FLOWS
C
      VOLIN(1)=0.0
      DO 500 K=2,KMAX
 500  VOLIN(K)=VOLIN(K-1)+DTH*(FLOW(K)+FLOW(K+1))



C       WRITE(CRT,2) (FLOW(K),K=1,KMAX)
C       WRITE(CRT,2) (VOLIN(K),K=1,KMAX)
C       PAUSE'PAUSE TO LOOK AT VOLIN(K) IN SUB FLOWW'

      RETURN
      END
C-----------------------------------------------------------------------
      SUBROUTINE TRANMPR(OMEGA,DPFG)
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),ZO(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),CPRL(15),DMAX(25),VO(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAXO,SLOPEO,
     + JSTADYN,JQSTATIC,DPGRAVO(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
       DIMENSION SUMDPG(10),II(10)
      INTEGER CRT
      CRT=0
  1   FORMAT(1X,9E8.2)
  2   FORMAT(3X,2I4,F8.2)
C        PAUSE'PAUSE AFTER YOU ENTER TRANSMP'
      IF(K.NE.100) GOTO 200
      KKK=300
  200 CONTINUE
      JmaxM1 = Jmax-1
      IF (K.EQ.1) GO TO 4
      IF (JQSTATIC.EQ.1) GO TO 8
      IF (JSTADYN.EQ.1) GO TO 8
C
C  FOR THE STATIC CASE PTM = PARENCHYMAL RECOIL  PRESSURE
C
  4   CONTINUE
      DO 6 J=4,7

      DO 5 N=1,NGENDP
  5   PTM(J,N)=P(J)
C        WRITE(CRT,1) (PTM(J,N),N=1,NGENDP)
C        PAUSE'PAUSE TO LOOK AT PTMs'
  6   CONTINUE
      GOTO 100
  8   CONTINUE
C        WRITE(CRT,2)JCYCLE
C        PAUSE'PAUSE TO LOOK AT JCYCLE'
      IF(JCYCLE.LT.0) GO TO 10
```

Program SBBOLUSD.FOR

```
C
C                   DYNAMIC CASE
C
C          SET XKM=1 AND XK=1 DURING INHALATION
C
      DO 9 J=4,7
      XKM1(J)=1.00
      DO 9 N=1,NGENDP
    9 XK(J,N)=1.0
      XK(2,2)=1.0
      XK(3,2)=1.0
      GO TO 75
   10 CONTINUE
C
C     CALCULATION OF XKM1=KM-1. Km IS A FUNCTION OF THE LOCAL
C     COMPLIANCE OF THE PARENCHYMA. SEE EQ.3.16.
C
      DO 50 J=4,7
      RFV=V(J)/RTLC(J)
      JI=II(J)
      FR=RFV-FVt(JI)
      IF(FR)31,33,32
   31 JI=JI-1
      IF(FVt(JI).LE.RFV) GO TO 33
      GO TO 31
   32 IF(JI.GE.JmaxM1) GO TO 33
      JI=JI+1
      IF(FVt(JI).GE.RFV) GO TO 33
      GO TO 32
   33 II(J) = JI
      IF(JI.GE.Jmax) JI=JmaxM1
      C=(FVt(JI+1)-FVt(JI))/DPptm
      XKM1(J)=2.0/(1.0+OMEGA*C)
C
C    CALCULATION OF XK - THE MEASURE OF THE INTERDEPENDENCE
C    BETWEEN THE LOCAL PARENCHYMA AND THE BRONCHIAL WALL.
C
C                   GENS 3-NGENDP
C
      DO 40 N=3,NGENDP
      AWDUE=DMAX(N)*RFV**(1./3.)
      DSTAR = AWD(J,N)/AWDUE
      IF(DSTAR.GT.1.0) DSTAR = 1.0
      IF(DSTAR.LE..7) DSTAR = .7
   40 XK(J,N)=XKM1(J)*(1.25/(1.0+((DSTAR-.7)/.15)**2)-.25)+1.0

C        WRITE(CRT,1)        (XK(JJ,N),N=3,NGENDP)
C        PAUSE'PAUSE TO LOOK AT (XK(JJ,N),N=3,NGENDP)'
  50   CONTINUE
C
C                             GENERATION 2
C

      DO 60 J=2,3
      IF(J.EQ.2) JJ=5
      IF(J.EQ.3) JJ=6
      RFV=V(JJ)/RTLC(JJ)
      AWDUE=DMAX(N)*RFV**(1./3.)
      DSTAR = AWD(J,N)/AWDUE
      IF(DSTAR.GE.1) DSTAR = 1.0
      IF(DSTAR.LE..7) DSTAR = .7
      XK(J,2)=XKM1(JJ)*(1.25/(1.0+((DSTAR-.7)/.15)**2)-.25)+1.0
C        WRITE(CRT,2) XK(J,2)
   60 CONTINUE
C        PAUSE'PAUSE TO LOOK AT THE XKs'
```

```
  75  CONTINUE
C
C     CALCULATE THE TRANSMURAL PRESSURE ACTING ON EACH
C          BRONCHIAL WALL. SEE EQ 3.15
C
C              GENERATIONS 3-NGENDP
C
      N1=NGENDP-2
      DO 85 J=4,7
      SUMDPG(J)=0.0
      DO 80 N=NGENDP,3,-1
      SUMDPG(J)=SUMDPG(J)-DPG(J,N)
  80  PTM(J,N)=P(J)+JCYCLE/XK(J,N)*SUMDPG(J)
  85  CONTINUE
C
C   CALCULATE PTM IN GENS 1 AND 2 AND THE TRACHEA
C
      SUMDPG(5)=SUMDPG(5)-DPG(2,2)
      SUMDPG(6)=SUMDPG(6)-DPG(3,2)
      PTM(2,2)=P(5)+JCYCLE/XK(2,2)*SUMDPG(5)
      PTM(3,2)=P(6)+JCYCLE/XK(3,2)*SUMDPG(6)
      SUMDPG(5)=SUMDPG(5)-DPFG
      PTM(1,1)=P(5)+JCYCLE*SUMDPG(5)
      PTMTRACH=P(5)+JCYCLE*SUMDPG(5)
 100  CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE STATVOL(FClung,Jstatus)
C
C   THIS SUBROUTINE CALCULATES THE FOUR REGIONAL VOLUMES USING AN
C   ITERATION SCHEME (METHOD OF FALSE POSITION).  THE PROCEDURE
C   STARTS WITH A GUESS OF P(4), THE PRESSURE AT THE CG OF THE
C   HIGHEST REGION.  THE OTHER PRESSURES ARE CALCULATED USING DPRAVO
C   AND THE RELATIVE VOLUME EXPANSION OF THE ENTIRE LUNG.  THE REGIONAL
C   AND TOTAL VOLUMES ARE THEN CALCULATED. THE CALCULATED VOLUME IS THE
C   COMPARED TO THE REQUIRED VOLUME,WHICH IS THE ERROR. THE ITERATION
C   SCHEME PROCEEDS ONCE TWO GUESSES HAVE BEEN MADE.
C
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),Z0(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),VO(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAXO,SLOPEO,
     + JSTADYN,JQSTATIC,DPGRAVO(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
      DIMENSION FCR(7),VR(7),PR(7)
      INTEGER R,CRT
      CRT=0
   1  FORMAT(3X,I5,4F10.2)
   2  FORMAT(2X,I3,8E9.3)
   3  FORMAT(2X,8E10.3)
C
      IGUESS=1
      CHECKV=2.0     !The calculated volume must be within CHECKV cc's
C    .          of the actual volume.
C
C USE P-FVA (PRESS-FRACTIONAL VOLUME OF THE ACINUS) TO GET THE 1st GUESS
C          FOR P(4) (PRESSURE AT THE CG OF REGION 4), P1
```

```
C
      J=Jmax
100   J=J-1
      IF(FVt(J).GT.FCLUNG) GOTO 100
      PG1=PTMMIN+(J-1)*DPptm
      PGRAV=DPGRAV(4)+DPGRAV(5)+DPGRAV(6)
      PR(4)= PG1+PGRAV/2          !PG1 IS THE FIRST GUESS
C      WRITE(CRT,2) J,FVt(J),FCLUNG,PTMMIN,PTMMAX,DPptm,PR(4)
C      PAUSE'PAUSE TO LOOK AT J,FVt(J),FCLUNG,PTMMIN,DPptm,PR(4)'
C
C  CALCULATE THE PRESSURES IN THE THREE LOWER REGIONS
C
      DO 200 R=5,7
 200  PR(R) = PR(R-1)-DPGRAV(R-1)
C      WRITE(CRT,3) (PR(R),R=4,7)
C      PAUSE'PAUSE TO LOOK AT PR(R),R=4,7 1st GUESS'
C
C  CALCULATE THE LUNG VOLUME FOR THESE PRESSURES
C
      VOLCALC=0

      DO 300 R=4,7


      I=1
250   I=I+1
      IF (PR(R).GT.Pst(Jstatus,R,I)) GOTO 250

      UU=(PR(R)-Pst(Jstatus,R,I-1))
     *          /(Pst(Jstatus,R,I)-Pst(Jstatus,R,I-1))

 280  FCR(R)= FCMIN + (I-2)*DFC +UU*DFC



      VR(R)=FCR(R)*RTLC(R)
      VOLCALC = VOLCALC+VR(R)   !VOLCALC is calculated lung volume

C      WRITE(CRT,2) I,Pst(Jstatus,R,I),PR(R),FCR(R),RTLC(R),FCMIN,DFC,U,VR(R)
C      PAUSE'I,PST,PR(R),FCR(R),RTLC(R),FCMIN,DFC,U,VR(R)- 1st guess'
 300  CONTINUE
C      WRITE(CRT,3) V(1),VOLCALC,ERR1
C      PAUSE' V(1),VOLCALC,ERR1 AFTER IST GUESS'
C
C  OBTAIN THE NEXT GUESS FOR P(4), PG2
C
      ERR1=VOLCALC-V(1)        !ERR1 is the error for the 1st guess, P1
      IF(ERR1)400,500,450
 400  PG2=PG1+.25     !VOLCALC too small, make PG2 bigger than PG1
      GO TO 500
 450  PG2=PG1-.25     !VOLCALC too big, make PG2 less than PG1
C
C  CALCULATE THE PRESSURES IN THE THREE LOWER REGIONS
C
 500  PR(4)=PG2
      DO 550 R=5,7
 550  PR(R)=PR(R-1)-DPGRAV(R-1)
C      WRITE(CRT,3) (PR(R),R=4,7)
C      PAUSE'PAUSE TO LOOK AT PR(R),R=4,7 FOR NEXT GUESS'
C
C  CALCULATE THE REGIONAL AND TOTAL LUNG VOLUMES FOR THESE PRESSURES
C
      VOLCALC=0
      DO 700 R=4,7
      I=1
```

```
 650  I=I+1
      IF (PR(R).GT.Pst(Jstatus,R,I)) GOTO 650

      UU=(PR(R)-Pst(Jstatus,R,I-1))
     *        /(Pst(Jstatus,R,I)-Pst(Jstatus,R,I-1))

 690  FCR(R)= FCMIN + (I-2)*DFC + UU*DFC
      VR(R)=FCR(R)*RTLC(R)
      VOLCALC = VOLCALC+VR(R)
C     WRITE(CRT,2) I,Pst(Jstatus,R,I),PR(R),FCR(R),RTLC(R),FCMIN,DFC,UU,VR(R)
C     PAUSE'PAUSE TO SEE I,PST,PR(R),FC(R),RTLC(R),FCMIN,DFC,UU,VR(R)'
 700  CONTINUE
      ERR2=VOLCALC-V(1)
      AERR2=ABS(ERR2)
      IF(AERR2.LT.CHECKV) GOTO 1000
C     WRITE(CRT,1) IGUESS,PG1,PG2,VOLCALC,ERR2
C     PAUSE'IGUESS,PG1,PG2,VOLCALC,ERR2'
C
C  USE THE ITERATION SCHEME TO MAKE THE NEXT GUESS, PG3
C
 750  CONTINUE
      PG3=(PG1*ERR2-PG2*ERR1)/(ERR2-ERR1)
      PG1=PG2
      PG2=PG3
      ERR1=ERR2
      IGUESS=IGUESS+1
      IF(IGUESS.LT.25) GOTO 800
      PAUSE'IGUESS=25 IN STATVOL, TYPE CONTROL C AND GET OUT!!!'
      KKK=1
 800  GOTO 500
 1000  CONTINUE

       DO 1100 R=4,7
       P(R)=PR(R)
       RVE(R)=(VR(R)+RWEIGHT(R))/RMINVOL(R)
 1100  V(R)=VR(R)
       V(2)=V(4)+V(5)
       V(3)=V(6)+V(7)
       RETURN
       END
C-----------------------------------------------------------------
      SUBROUTINE AIRWAY(TRCDMAX,TRCDIAM)
C
C THIS SUBROUTINE CALCULATES THE AIRWAY LENGTHS AND DIAMETERS DURING THE
C BREATH.  LENGTHS ARE BASED ON THE REGIONAL EXPANSION AND DIAMETERS ARE
C  BASED ON THE TRANSMURAL PRESSURE THAT ACTS ON THE AIRWAY.
C
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),Z0(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),V0(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DO,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAX0,SLOPE0,
     + JSTADYN,JQSTATIC,DPGRAV0(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)

       INTEGER CRT,R
       CRT=0
  1    FORMAT(2X,8E9.3)
  2    FORMAT(2X,I5)
  3    FORMAT(2X,I4,4F9.3)
```

Program SBBOLUSD.FOR

```
  4    FORMAT(2X,2I5,F10.4)
C
C    CALCULATE THE LENGTHS FOR GENS 1 AND 2
C
C      WRITE(CRT,2) R
      AWL(1,1) = (V(1)/RTLC(1))**(1./3.) * XLMAX(1)
      AWL(2,2) = (V(2)/RTLC(2))**(1./3.) * XLMAX(2)
      AWL(3,2) = (V(3)/RTLC(3))**(1./3.) * XLMAX(2)
C      PAUSE'IN AIRWAY, AFTER AWL(3,2)'
C
C    CALCULATE THE LENGTHS FOR GENS 3-23
C
      DO 100 R=4,7
      SCALE=(V(R)/RTLC(R))**(1./3.)
C      WRITE(CRT,3) R,V(R),RTLC(R),SCALE
      DO 50 N=3,NGENDP
 50   AWL(R,N)=SCALE*XLMAX(N)
C     WRITE(CRT,2) R
C     WRITE(CRT,1) (AWL(R,N),N=3,NGENDP)
C     PAUSE'PAUSE TO LOOK AT THE AWLS'
 100  CONTINUE
C
C    CALCULATE THE AIRWAY DIAMETERS FOR THE TRACHEA AND GEN 1 (AREA 2)
C
       TRCDIAM= 0.80+0.056*PTM(1,1) * TRCDMAX
       AWD(1,1)=0.82+0.050*PTM(1,1) * DMAX(1)
C
C    CALCULATE THE AIRWAY DIAMETERS FOR GEN's 2-NGENDP
C
      PTM(4,2)=PTM(2,2)    !calculate PTM(j,2) for use in the do loop
      PTM(5,2)=PTM(2,2)
      PTM(6,2)=PTM(3,2)
      PTM(7,2)=PTM(3,2)
      if(k.LT.237) GOTO 119
      KKK=1
 119  CONTINUE

      DO 200 R=4,7
      DO 160 N=2,NGENDP
      IF(PTM(R,N).LT.0.0) GOTO 150

C   CALCULATION OF AIRWAY DIAMETERS FOR POSITIVE PTM's
      I=1
 120  I=I+1
      PTT=PTMMIN+(I-1)*DPptm
      IF(I.EQ.JmaxPTM) GOTO 130
      IF(PTM(R,N).GT.PTT) GOTO 120
      UU=(PTM(R,N)-(PTT-DPptm))/DPptm
      SCALE=FDMAX(I-1)+UU*(FDMAX(I)-FDMAX(I-1))
      GOTO 140
 130  UU=(PTM(R,N)-(PTT-DPptm))/DPptm
      SCALE= FDMAX(JmaxPTM)+UU*(FDMAX(JmaxPTM)-FDMAX(JmaxPTM-1))
 140  AWD(R,N)=SCALE*DMAX(N)
      GOTO 160

C   CALCULATION OF AIRWAY DIAMETERS FOR NEGATIVE PTM'S
 150  AWD(R,N)=FDMAX0*DMAX(N)/(1.-PTM(R,N)/SLOPE0)**(1./3.)

 160  CONTINUE
      IF(K.LT.1000) GOTO 200
      WRITE(CRT,2) R
      WRITE(CRT,1) (AWD(R,N),N=2,NGENDP)
      WRITE(CRT,1) (PTM(R,N),N=2,NGENDP)
      PAUSE'PAUSE TO LOOK AT THE AWDS'
 200  CONTINUE
```

```
      AWD(2,2)=AWD(5,2)
      AWD(3,2)=AWD(6,2)
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE PRESSLS(A1,A2,LSIT,QLATEST)
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),ZO(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),VO(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAXO,SLOPEO,
     + JSTADYN,JQSTATIC,DPGRAVO(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
C      LSIT=1 CALCULATE TOTAL AIRWAY AREAS AND PRESSURES FOR ALL
C           AIRWAYS FROM NGENDP TO GENERATION 2 (1st CALL FROM MAIN)
C      LSIT=2 CALCULATE PRESSURE CHANGES IN ALL AIRWAYS (LATER CALLS
C           FROM PROGRAM MAIN
C      LSIT=3 CALCULATE PRESSURE CHANGES IN AREAS A1 AND A2 WHEN
C           CALLED FROM REGFLOW
C
      INTEGER CRT,AREA,A1,A2
      CRT=0
    1 FORMAT(2X,7E10.3)
    2 FORMAT(5X,3I10)
C      WRITE(CRT,2)LSIT,NGENDP,NGENDP1
C      PAUSE'PAUSE AFTER YOU ENTER PRESSLS AND SHOW LSIT,NGENDP,NGENDP1'
C
      PIE=3.1416
      PIE4=PIE/4
      SQ2PI= 1.4142*PIE*VISCSTY
      NGENDP1=NGENDP+1               !calculates velocity for an extra gen
      IF(LSIT.NE.1)GOTO 210     !areas already calculated
C
C     CALCULATE THE CROSS-SECTIONAL AREA OF THE AIRWAYS IN EACH
C          GENERATION OF EACH OF THE AREAS
C
C        CROSS-SECTIONAL AREA OF GENERATIONS 3-NGENDP
C
      DO 150 AREA=A1,A2
      DO 100 N=3,NGENDP
      NAWPG=(2**N)/4    !# of Airways Per Generation in areas 4,5,6 and 7
      TXAR(AREA,N)=NAWPG*PIE4*AWD(AREA,N)**2
  100 CONTINUE
      IF(K.LT.1000) GOTO 150
      WRITE(CRT,2) AREA
      WRITE(CRT,1) (TXAR(AREA,J),J=3,NGENDP1)
      PAUSE'PAUSE TO LOOK AT THE AREAs OF GENs 3-21'
  150 CONTINUE
C
C     AREA OF GENERATION 2
C
       NAWPG=2               !Number of Airways in gen 2 in AREAs 2 and 3
       N=2
      DO 200 AREA=2,3
      TXAR(AREA,N)=NAWPG*PIE4*(AWD(AREA,N))**2
C     WRITE(CRT,1) TXAR(AREA,2)
C     PAUSE'PAUSE TO LOOK AT THE AREA IN SECOND GENERATION'
  200 CONTINUE
```

```
 210  CONTINUE
C
C       CALCULATE THE GAS VELOCITIES IN THE AIRWAYS
C   NOTE: THE GAS VELOCITIES IN THE 1st GEN AND THE TRACHEA ARE NOT
C       REQUIRED FOR THE ITERATION.  THESE ARE CALCULATED IN THE
C       MAIN PROGRAM TO LOOK AT OVERALL PRESSURES
C
C  FIRST CALCULATE THE FLOWS TO ALL AREAS BASED ON THIS GUESS-THE
C  VARIABLE USED IS QR.  IT IS USED ONLY IN PRESSLS AND REGVOL TO
C  SAVE THE FLOWS BASED ON THE LATEST GUESS FOR Q(2) FROM PROG MAIN
C
      IF(LSIT.EQ.3) GOTO 220
      QR(2)=QLATEST
      QR(3)=Q(1)-QR(2)
      QR(4)=Z(4)*2*QR(2)
      QR(5)=QLATEST-QR(4)
      QR(6)=Z(6)*2*QR(3)
      QR(7)=QR(3)-QR(6)
      GOTO 240
 220  CONTINUE
      QR(A1)=QLATEST           !need flows for A1 and A2 when called from REGFLOW
      J=A2/2
      QR(A2)=QR(J)-QR(A1)
 240  CONTINUE
C      WRITE(CRT,1) (QR(AREA),AREA=A1,A2)
C      PAUSE'PAUSE TO LOOK AT THE REGIONAL FLOWS IN PRESSLS'
C
C         CALCULATE THE GAS VELOCITIES IN GEN 3-NGENDP
C
      DO 275 AREA=A1,A2
      DO 250 N=3,NGENDP
      U(AREA,N)=QR(AREA)/TXAR(AREA,N)
 250  CONTINUE
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) (U(AREA,J),J=3,NGENDP1)
C     PAUSE'PAUSE TO LOOK AT THE VELOCITIES IN GENs 3-21'
 275  CONTINUE
      IF(LSIT.EQ.3) GOTO 310
C
C     GAS VELOCITIES IN GEN 2 IN AREAS 2 AND 3
C
      N=2
      DO 300 AREA=2,3
      U(AREA,N)=QR(AREA)/TXAR(AREA,N)
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) QR(AREA),U(AREA,2)
C     PAUSE'PAUSE TO LOOK AT QR AND U FOR THE SECOND GENERATION'
 300  CONTINUE
C
C      GAS VELOCITY IN GENERATION 1
C
      U(1,1)=ABS(Q(1)/(1.571*AWD(1,1)**2))
C
 310  CONTINUE
C
C   CALCULATE THE VISCOUS LOSSES IN THE FLOW USING A CORRECTED EQ 3.26
C     DPV IS NEGATIVE FOR INHALATION;POSITIVE FOR EXHALATION
C     INHALATION FLOWS PRODUCE POSITIVE VELOCITIES
C     EXHALATION FLOWS PRODUCE NEGATIVE VELOCITIES
C
C          VISCOUS LOSSES IN GENERATIONS 3-NGDP
C
      DO 400 AREA=A1,A2
      DO 350 N=3,NGENDP
      ABVEL=ABS(U(AREA,N))
```

```
      RE=ABVEL*AWD(AREA,N)/VISK
      W1=(32.0*AWL(AREA,N))/(RE*AWD(AREA,N))
      W2=DENSITY*U(AREA,N)**2
      POIS(AREA,N)=GI*W1*W2
      W3=((RE*AWD(AREA,N)/AWL(AREA,N))**(1./2.))/3.2703
      POIS(AREA,N)=W3*POIS(AREA,N)
      SQRE=(RE*AWD(AREA,N)/AWL(AREA,N))**(1./2.)
      COEFB=-U(AREA,N)*AWL(AREA,N)/AWD(AREA,N)**2
      DPV(AREA,N)=COEFA*COEFB*SQRE
 350  DPV(AREA,N)=GI*DPV(AREA,N)   !convert DP from dynes/cm**2 to cmH2O
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) (DPV(AREA,N),N=3,NGENDP)
C     PAUSE'PAUSE TO LOOK AT THE DPVs IN GENs 3-NGENDP'
 400  CONTINUE
      IF(LSIT.EQ.3) GOTO 510
C
C       VISCOUS LOSSES IN GENERATION 2
C
      N=2
      DO 500 AREA=2,3
      ABVEL=ABS(U(AREA,N))
      RE=ABVEL*AWD(AREA,N)/VISK
      SQRE=(RE*AWD(AREA,N)/AWL(AREA,N))**(1./2.)
      COEFB=-U(AREA,N)*AWL(AREA,N)/AWD(AREA,N)**2
      DPV(AREA,N)=COEFA*COEFB*SQRE
      DPV(AREA,N)=GI*DPV(AREA,N) !convert DP from dynes/cm**2 to cmH2O
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) RE,DPV(AREA,N)
 500  CONTINUE
C     PAUSE'PAUSE TO LOOK AT RE AND DPVs IN GEN 2'
 510  CONTINUE
C
C  CALCULATE THE CONVECTIVE ACCELERATIONS USING EQ 3.28
C     DPC IS POSITIVE FOR INHALATION;POSITIVE FOR EXHALATION
C
      DO 600 AREA=A1,A2
      DO 550 N=3,NGENDP
      DPC(AREA,N)=DK*(U(AREA,N-1)**2-U(AREA,N)**2)
 550  DPC(AREA,N)=GI*DPC(AREA,N)!convert dynes/cm**2 into cmH2o
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) (DPC(AREA,N),N=3,NGENDP)
C     PAUSE'PAUSE TO LOOK AT THE DPC FOR GENs 3-NGENDP'
 600  CONTINUE
      IF(LSIT.EQ.3) GOTO 710
C
C           CONVECTIVE ACCELERATIONS IN GENERATION 2
C
      N=2
      DO 700 AREA=2,3
      DPC(AREA,N)=DK*(U(AREA,N-1)**2-U(AREA,N)**2)
      DPC(AREA,N)=GI*DPC(AREA,N)!convert dynes/cm**2 into cmH2o
C     WRITE(CRT,2) AREA
C     WRITE(CRT,1) DPC(AREA,N)
 700  CONTINUE
C     PAUSE'PAUSE TO LOOK AT DPC FOR GEN 2'
 710  CONTINUE
C
C   CALCULATE THE TOTAL PRESSURE CHANGE IN EACH REGION (DPRL)
C             AND IN GENERATION 2 (DPG)
C
C                     CALCULATION OF DPR
C
      DO 800 AREA=A1,A2
      DPR(AREA)=0.0
      DO 750 N=3,NGENDP
```

Program SBBOLUSD.FOR

```
      DPG(AREA,N)=DPV(AREA,N)+DPC(AREA,N)  !DPG=DP in a gen
 750  DPR(AREA)=DPR(AREA)+DPG(AREA,N)
 800  CONTINUE
C      WRITE(CRT,1) (DPR(AREA),AREA=A1,A2)
C      PAUSE'PAUSE TO LOOK AT THE DPR CALCULATED IN PRESSLS'
      IF(LSIT.EQ.3) GOTO 910
C
C           CALCULATION OF DPG FOR GENERATION 2
C
      N=2
      DO 900 AREA=2,3
      DPG(AREA,2)=DPV(AREA,N)+DPC(AREA,N)
C      WRITE(CRT,2) AREA
C      WRITE(CRT,1) DPG(AREA,N)
 900   CONTINUE
C      PAUSE' PAUSE TO LOOK AT DPGs FOR AREAS 2 AND 3 CALC IN PRESSLS'
 910  CONTINUE
      RETURN
      END
C-------------------------------------------------------------------
      SUBROUTINE  REGFLOW(CHECKP)
       COMMON FLOW(601),VOLIN(601),F(15),RTLC(15),WL(25),WD(25),
     + P1(201),V(7),AWL(7,18),H(7),Z(10),Z0(10),RCV(7),DVdt,COEFA,
     + AWD(7,18),Q(7),Pst(2,7,151),PTM(7,30),DENSITY,VISCSTY,VISK,
     + P(15),DPR(15),RES(15),DPRL(15),DMAX(25),V0(15),JCOUNT,U(7,20),
     + VOLD(15),K,ImaxFC,NUNITS,DFC,FCMIN,APGPL(25),NGEN,DD,XLMAX(24),
     + PERIOD,DT,TMAX,FCMAX,DELVOL,FDMAX(201),XKM1(7),XK(7,25),
     + DPG(7,18),RESCON(7,18),RESVIS(7,18),JINSIGN,JEXSIGN,FCFRC,TLC,
     + DPptm,ImaxFCM1,JCYCLE,PTMMIN,PTMMAX,VE,VMINLUNG,WEIGHT,DPGRAV(7),
     + FVt(201),ImaxFCP1,Jmax,TXAR(7,30),POIS(7,30),FDMAX0,SLOPE0,
     + JSTADYN,JQSTATIC,DPGRAV0(7),DFCS,DFC2,AMOUNT(7),GI,JmaxPTM,
     + NGENDP,NGENDP1,DK,QG(5),RWEIGHT(10),RVE(10),RMINVOL(10),QR(10),
     + DPV(7,30),DPC(7,30)
C
C  THIS SUBROUTINE CALCULATES THE FLOWS WITHIN AREAS 2 AND 3 FOR A GIVEN
C  Q(2) AND Q(3) THAT HAVE BEEN DEFINED IN THE MAIN PROGRAM
C
C
      INTEGER AREA,A1,A2,CRT
   1  FORMAT(2X,7E10.3)
   2  FORMAT(5X,3I5)

      CRT=0
      LSIT=3
C     PAUSE'PAUSE AFTER YOU ENTER REGFLOW'
C     WRITE(CRT,1) P(4),P(5),P(6),P(7)
C     PAUSE'PAUSE AFTER YOU PRINT YOUR Ps'
C
      DO 1000 AREA=2,3
      A1=2*AREA                        !A1 is AREA 4 or 6
      A2=A1+1                          !A2 is AREA 5 or 7
C      WRITE(CRT,2) A1,A2,AREA
C      PAUSE'PAUSE TO LOOK AT A1,A2,AREA AS YOU ENTER REGFLOW'
C
C  THE ITERATION DETERMINES THE FLOW IN AREA A1.  NEED TWO GUESSES FOR
C  THE FLOW IN AREA A1 TO GET THE ITERATION PROCESS (METHOD OF FALSE
C  POSITION) STARTED.  FIRST GUESS IS Q2 AND IT IS BASED ON THE LATEST
C  GUESS FOR Q(2) FROM PROG MAIN.  QRs ARE CALCULATED IN PRESSLS FOR
C  THE CASE LSIT=1.
C
      Q2=QR(A1)
C
C  NOTE:THE DPRs FOR THIS GUESS WERE CALCULATED JUST BEFORE THE MAIN
C  PROGRAM ENTERED REGFLOW.  THE ITERATION FUNCTION (F)IS THE ERROR
C  WHEN EQ. 3.32 IS NOT PROPERLY BALANCED.
```

Program SBBOLUSD.FOR

```
C
      G=P(A1)-P(A2)-DPGRAV(A1)
      F2=G+(DPR(A2)-DPR(A1))
      AF2=ABS(F2)
      IF(AF2.LT.CHECKP)GOTO 500
C
C     GET SECOND GUESS FOR FLOW TO AREA A1
C
      IF(F2.GT.0.00) Q3=1.02*Q2
      IF(F2.LT.0.00) Q3=0.98*Q2
C      WRITE(CRT,1) Q2,Q3,F2
C      PAUSE'PAUSE TO LOOK AT Q2,Q3,F2 FROM THE FIRST GUESS FOR Q(A1)'
      LCOUNT=1
 200  F1=F2
      Q1=Q2
      Q2=Q3
      CALL PRESSLS(A1,A2,LSIT,Q2)
      F2=G+(DPR(A2)-DPR(A1))
      AF2=ABS(F2)
      LCOUNT=LCOUNT+1
C      WRITE(CRT,2) LCOUNT
C      WRITE(CRT,1) Q1,Q2,F1,F2
C      PAUSE'PAUSE AFTER Q1,Q2,F1,F2'
      IF(AF2.LT.CHECKP) GOTO 500
C
C     GET NEXT VALUE OF FLOW TO AREA A1
C
      Q3=(Q1*F2-Q2*F1)/(F2-F1)
      GOTO 200
  500 CONTINUE
      Q(A1)=Q2
      Q(A2)=QG(AREA)-Q2
C      WRITE(CRT,2) A1,A2
C      WRITE(CRT,1) Q(A1),Q(A2)
C      PAUSE'PAUSE AFTER A1,A2,Q(A1,Q(A2): ITERATION FINISHED!!!'
 1000 CONTINUE
      RETURN
      END
```

Appendix C - 4
Convection-Diffusion Model

PROGRAM CONDIF

```
      DIMENSION    C(4,501),Calv(4,501),
     * C0(101),Cso(101),Qsac(101),Q(151),
     * H(101),F(101),THETA1(101),THETA2(101),
     * G(101),ALPHA1(101),ALPHA2(101),BETA1(101),BETA2(101),
     * B1(101),B2(101),B3(101),B4(101)

      COMMON AVTO,A0,DT,DTH,DZ,TIDVOL,VLT,TL,TD,A(101),
     * AV(101),VZ(101),AVZ(201),FA(201),FAZ(201),VOW,CC(151),
     * FLOW(501),VOLIN(501),Z(4,501),Rbeg(4),RV0(4),RV(4),
     * IMAX1,TV,AWL(101),AWD(101),Aso(101),V(151),Falv(151),
     * Vsa(151),SD,Ctrach0

6     FORMAT (10X,8E13.5)

      IMAX=101          !Number of discrete points on the gen coordinate sys
      IMAX1=IMAX-1

      CALL FLOWW(KMAX,KEINS,VSTART)
      CALL GEOM(VSTART,IMAX)

C     INITIALIZE VARIABLES

      DO 20 I=1,IMAX
      C(L,I)=0.78
      C0(0)=0.78
20    Calv(L,I)=0.78
      Ctrach0=0.78

      Cin=0.0                    !Concentration of N2 in the inspired gas
      Dm=0.25                    !Molecular diffusion coefficient for N2
      FLUX=0.0
      CMOUTH=1.0
      DZS=DZ**2
      DZH=DZ/2.
      T=0.0
      U=0
      P=DZS/DT

      DO 100 I=1,IMAX
      H(I)=A(I)/AWL(I)
      F(I)=A(I)*AWL(I)
      THETA1(I)=2*Aso(I)*Dm*((A(I)/2+SD)*Vsa(I))
100   THETA2(I)=DT/(2.0*Vsa(I))


      DO 2000 K=2,KMAX

      T=T+DT
      CMOUTH0=CMOUTH
      Q0=FLOW(K-1)
      DO 150 L=1,4
      C(L,1)=0.0
      IF (VOLIN(K).GT.TV) C(L,1)=Cin

150   CONTINUE
      D1= 0.37*ABS(FLOW(K))*AWD(1)/VZ(1)/DZ
      IF (Q0.GT.0.0.AND.K.GE.KEINS)
     *            CALL ENTRAN(K,KEINS,Q0,CMOUTH,C(L,1),FLUX)

      DO 1000 L=1,4
```

Program CONDIF.FOR

```
C  S IS THE LINEAR SCALING FACTOR BETWEEN THE PREENT VOLUME OF REGION L
C  AND WEIBEL'S ANATOMICAL LUNG MODEL DATA.

      IF(K.GT.2) GOTO 170
      S=(Rbeg(L)/VOW)**(1./3.)
      S2=S**2
      S3=S**3
 170  CONTINUE

      S0=S
      S02=S2
      S03=S3

      Vreg=RVO(L)+.5*(Z(L,K-1)*FLOW(K-1)*Z(L,K)*FLOW(K))
      S=((4.0*Vreg)/VOW)**(1./3.)
      S2=S**2
      S3=S**3

      DO 200 I=1,IMAX1
      CO(I)=C(L,I)
      Cso(I)=(CO(I)+Calv(L,I))/2.0
      Qsac(I)=Z(L,K)*FLOW(K)*(1.-Falv(I))
      Q(I)=Z(L,K)*(1- FA(I))
 200  CONTINUE

       DO 300 I=2,IMAX1
       G(I)= S*(Dm*H(I-1)+Dm*H(I))/2.0
       ALPHA1(I)=S0*Cso(L)/(S3-THETA2(I)*Qsac(I)+S*THETA1(I))
       ALPHA2(I)=(THETA2(I)*Qsac(I)-S*THETA1(I))/
     *      (S3-THETA2(I)*Qsac(I)+S*THETA1(I))
       BETA1(I)=Vsa(I)/DZ*(S3*ALPHA1(I)-S03*Cso(I))
 300   BETA2(I)=Vsa(I)/DZ*(S3*ALPHA2(I))

C   CALCULATE THE COEFFICIENTS TO THE DIFFERENTIAL EQ

      DO 400 I=2,IMAX1
      B1(I)=-(G(I)+G(I-1))/2.0-DZH*Q(I-1)
      B2(I)=P*(S3*F(I)+BETA2(I))+(G(I)+G(I+1))/2+(G(I-1)+G(I))/2.0
      B3(I)=-(G(I)+G(I+1))/2.0+DZH*Q(I+1)
 400  B4(I)=P*(S0*F(I)*CO(I)-BETA1(I))

C  SET THE BOUNDARY CONDITIONS AN I=1 AND AT I=IMAX

      B1(1)=0.0
      B2(1)=1.0
      B4(1)=C(L,1)
      IF (QO.LT.0.) B1(1)=1.0
      IF (QO.LT.0.) B4(1)=FLUX/D1
      B2(IMAX)=1.0
      B3(IMAX)=1.0
      B4(IMAX)=0.0

C  SOLVE THE TRIDIAGONAL MATRIX TO GET THE CONCENTRATIONS ALONG THE
C  THE AIRWAY WITHIN THIS REGION (C(L))

       DO 500 I=1,IMAX
 500   CC(I)=C(L,I)

       CALL SOLVE(B1,B2,B3,B4,IMAX,CC)

      DO 600 I=1,IMAX
 600  C(L,I)=CC(I)

      IF (QO.GT.0.0) CALL ENTRAN(K,KEINS,QO,CMOUTH,C(L,1),FLUX)
 1000 CONTINUE
```

Program CONDIF.FOR

```
 2000  CONTINUE
       STOP
       END
C-----------------------------------------------------------------
      SUBROUTINE GEOM(VSTART,IMAX)

C THIS SUBROUTINE CALCULATES THE ANATOMICAL VARIABLES THAT ARE USED IN
C          THE CONVECTION/DIFFUSION MODEL

       COMMON AVT0,A0,DT,DTH,DZ,TIDVOL,VLT,TL,TD,A(101),
     * AV(101),VZ(101),AVZ(201),FA(201),FAZ(201),VOW,CC(151),
     * FLOW(501),VOLIN(501),Z(4,501),Rbeg(4),RV0(4),RV(4),
     * IMAX1,TV,AWL(101),AWD(101),Aso(101),V(151),Falv(151),
     * Vsa(151),SD,Ctrach0

      DIMENSION Valv(151),Svalv(151),SvalvZ(151),Sawl(151),SawlZ(151)

C   INPUT THE ANATOMICAL DATA

      TDW=1.8
      TLW=22.0

       READ(5,1) (AWL(N),N=1,23)
       READ(5,1) (AWD(N),N=1,23)
  1    FORMAT(13F6.3)
  2    FORMAT(7F11.0)

       DO 10 N=1,16
 10    Valv(N)=0.0

       READ(5,1) (Valv(N),N=17,23)!Alveolar volume per generation

       SD=0.032                !Sac Depth in cm
       Valunit=0.00002671      !Volume of an ALveolar UNIT cm3
       Asop= 0.00052           !Area of a Sac OPening

C  INTERPOLATE THE ANATOMICAL DATA FROM THE 23 GERNERATION GRID
C          TO A FINER GRID OF IMAX POINTS.

      FIMAX1=IMAX1
      DZ=23./FIMAX1

      Aso(1)=0.0
      SValv(1)=0.0
      SVcaw=0.0
      V(1)=0.0
      A(1)=0.0
      Sawl(1)=0.0

      DO 20 N=2,24
      Sawl(N)=Sawl(N-1)+AWL(N)!Summed lungth of conductin airways
      FNUM=(2**N)/4.0          !# of airways/gen in a region
      A(N)=FNUM*3.1416*(AWD(N)**2)/4.!Cross-sectional area of con aw
      SVcaw=SVcaw+A(N)*AWL(N) !Summed vol of con airways
      SValv(N)=SValv(N-1)+Valv(N)!Summed vol of alveoli
 20   V(N)=SValv(N)+SVcaw     !Summed vol of alveoli and airways

      DO 25 N=1,24
      Falv(N)=Valv(N)/V(24)   !Alveolar vol/gen as a fraction of tot vol
 25   FA(N)=V(N)/V(24)        !Fraction of the total volume

      CALL DERIV(FA,24,1.0,FAZ)
      CALL DERIV(SValv,24,1.0,SValvZ)
      CALL DERIV(Sawl,24,1.0,SawlZ)
      CALL DERIV(V,24,1.0,VZ)
```

```
      DO 28 N=1,24
      IF (N.LT.17) SValvZ(N)=0.0
28    IF (FA(N).GT.1.0) FA(N)=1.0

      DO 50 N=2,24
      Nsac=SValvZ(N)/(4.0*Valunit) !# of sacs per generation in a region
      Aso(N)=Asop*Nsac          !Regional Area of sac openings/generation
50    A(N)=VZ(N)/SawlZ(N)

      CALL INTERP(FA,24,1.0,IMAX,DZ)
      CALL INTERP(FAZ,24,1.0,IMAX,DZ)
      CALL INTERP(SValvZ,24,1.0,IMAX,DZ)
      CALL INTERP(SawlZ,24,1.0,IMAX,DZ)
      CALL INTERP(VZ,24,1.0,IMAX,DZ)
      CALL INTERP(A,24,1.0,IMAX,DZ)
      CALL INTERP(V,24,1.0,IMAX,DZ)

C     SCALE WEIBEL'S DATA TO SUBJECT'S LUNG STARTING LUNG VOLUME

      VOW=3545.
      SCALE=(VSTART/VOW)**(1./3.)
C
C     CALCULATE THE NEW TRACHEA GEOMETRIES
C
      TD=TDW*SCALE
      TL=TLW*SCALE
      A0=3.1416*TD**2/4.
      TV=TL*A0

99    FORMAT (11E11.3)
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE FLOWW(KMAX,KEINS,VSTART)

C    THIS SUBROUTINE READS IN THE FLOW AT THE MOUTH, AND THE REGIONAL
C    FLOWS AND VOLUMES THAT WERE CALCULATED IN THE VENTILATION MODEL

        COMMON AVT0,A0,DT,DTH,DZ,TIDVOL,VLT,TL,TD,A(101),
     * AV(101),VZ(101),AVZ(201),FA(201),FAZ(201),VOW,CC(151),
     * FLOW(501),VOLIN(501),Z(4,501),Rbeg(4),RV0(4),RV(4),
     * IMAX1,TV,AWL(101),AWD(101),Aso(101),V(151),Falv(151),
     * Vsa(151),SD,Ctrach0

      INTEGER CRT
      CRT=0
 1    FORMAT(I3,F7.2,F7.1)
 2    FORMAT(9F8.1)
 3    FORMAT(3X,I5)

C    READ THE FLOW DATA FROM THE DATA FILE CALLED FLOW.DAT
C         JMAX = # OF FLOW DATA POINTS FROM EXPERIMENTAL RECORD
C         VSTART = GAS VOLUME OF LUNG AT START OF INHALATION
C         DTC = TIME INCREMENT FOR MEASURED FLOW DATA
C         TIDVOL = MEASURED TIDAL VOLUME
C         RV(L)= REGIONAL VOLUMES FOR REGIONS L=1,4
C         FLOW(J) = MEASURED FLOWS AT THE MOUTH
C         Z(L,J) = FRACTION OF FLOW GOING TO EACH REGION L AT TIME J

C       INPUT FLOW DATA
C
      NI5=5
C    OPEN INPUT FILE 'EXPFLOW.DAT'
      OPEN(NI5,FILE='EXPFLOW.DAT')
      READ(NI5,1) JMAX,DTC,TIDVOL
```

Program CONDIF.FOR

```
      READ(NI5,2) (RV(L),L=1,4)

      VSTART=RV(1)+RV(2)+RV(3)+RV(4)
      DO 40 L=1,4
 40   Rbeg(L)=RV(L)       !Set beginning volumes for each Region L

      READ(NI5,2) (FLOW(J),J=1,JMAX)
      DO 50 L=1,4
 50   READ(NI5,2) (Z(L,J),J=1,JMAX)

C      WRITE(CRT,1) JMAX,VSTART,DTC,TIDVOL
C      WRITE(CRT,2) (FLOW(J),J=1,JMAX)
C      PAUSE'PAUSE TO LOOK AT JMAX,VSTART,DTC,TIDVOL AND FLOWS'
C
C      DETERMINE WHERE THE MEASUED DATA BECOMES NEGATIVE, JEINS

      J=0
 100  J=J+1
      IF(FLOW(J).GE.0.0) GOTO 100
      JEINS=J
C
C   INTERPRET FLOW DATA TO A FINER KMAX GRID.  THE TIME STEP FOR THIS
C   IS BASED ON AN AVERAGE VOLUME INCREMENT OF DELVOL DURING THE INHALATION
C   PART OF THE BREATH.

      JMAX1=JMAX-1
      TMAX=DTC*JMAX1
      DELVOL=20.0  !average volume increment in cc's for time step DT
      TIMEINH = (JEINS-1)*DTC
      DT=TIMEINH*(DELVOL/TIDVOL)
      KMAX1=TMAX/DT
      DT=TMAX/KMAX1
      KMAX=KMAX1+1
      DTH=DT/2.0

      CALL INTERP(FLOW,JMAX,DTC,KMAX,DT)

      WRITE(CRT,2) (FLOW(I),I=1,KMAX)
      WRITE(CRT,2) DTC,DT
      PAUSE'PAUSE TO LOOK AT FLOWS AND DTC AND DT IN SUB FLOWW'
C
C DETERMINE WHEN THE FLOW FIRST BECOMES NEGATIVE, THE END OF INHALATION.
C         LOOK FOR KCHECK FLOWS LESS THAN ZERO.
C
      KCHECK=3
      KC=0
      K=0
 150  K=K+1
C      WRITE(CRT,3)K
      IF(FLOW(K).GT.0) GOTO 200
      KC=KC+1
      IF(KC.EQ.KCHECK) GOTO 250
      GOTO 150
 200  KC=0
      GOTO 150
 250  KEINS=K-(KCHECK-1)
C      PAUSE'PAUSE TO LOOK AT ALL THE Xs'

C  INTEGRATE FLOWS TO GET TIDAL VOLUME BASED ON FLOW DATA (FTIDVOL)

      KEINS1=KEINS-1
      FTIDVOL=0.0
      DO 300 K=1,KEINS1
 300  FTIDVOL=FTIDVOL+DTH*(FLOW(K)+FLOW(K+1))
```

```
C     NORMALIZE THE FLOWS WITH RESPECT TO THE MEASURED TIDAL VOLUME

      SCALE=TIDVOL/FTIDVOL
      DO 400 K=1,KMAX
 400  FLOW(K)=SCALE*FLOW(K)

C    CALCULATE THE INHALD VLUME (VOLIN) BASED ON THE SCALED FLOWS

      VOLIN(1)=0.0
      DO 500 K=2,KMAX
 500  VOLIN(K)=VOLIN(K-1)+DTH*(FLOW(K-1)+FLOW(K))

C      WRITE(CRT,2) (FLOW(K),K=1,KMAX)
C      WRITE(CRT,2) (VOLIN(K),K=1,KMAX)
C      PAUSE'PAUSE TO LOOK AT FLOW(K) AND VOLIN(K) IN SUB FLOWW'

      RETURN
      END
C-----------------------------------------------------------------------
      SUBROUTINE ENTRAN(K,KEINS,QO,CMOUTH,CTRACH,FLUX)
         COMMON AVTO,AO,DT,DTH,DZ,TIDVOL,VLT,TL,TD,A(101),
     * AV(101),VZ(101),AVZ(201),FA(201),FAZ(201),VOW,CC(151),
     * FLOW(501),VOLIN(501),Z(4,501),Rbeg(4),RVO(4),VR(4),
     * IMAX1,TV,AWL(101),AWD(101),Aso(101),V(151),Falv(151),
     * Vsa(151),SD,Ctrach0

      DIMENSION C(201),CO(201),SDC(201)

      AQO=ABS(FLOW(K))
      RET=AQO*TD/AO/0.15
      FK=0.05/(2.14*ALOG(RET)-3.6)
      D=FK*TD*AQO/AO
      IMAX=21
      IMAX1=IMAX-1
      FIMAX1=IMAX1
      DX=TL/FIMAX1
      NT=1.+2.*D*DT/DX**2
      FNT=NT
      L=0.5+(AQO/AO)*DT/FNT/DX
      FL=L
      DX=(AQO/AO)*(DT/FNT)/FL
      DX2=DX**2
      IMAX=TL/DX+1.5
      IMAX1=IMAX-1
      P=D*(DT/FNT)/DX**2
      WRITE (6,18) NT,IMAX,L,DX,D,P,CMOUTH,CTRACH
18    FORMAT (5X,3I5,8E14.3)
      L1=L+1
      IML=IMAX-L
      IML1=IML+1
      NUMJ=KEINS
      DO 15 I=1,IMAX
15    C(I)=0.0
20    CONTINUE
      IQO=QO/AQO
      NCYC=1+(K-1)/NUMJ
      JCYC1=1+(NCYC-1)+NUMJ
      IF (K.EQ.JCYC1) C(1)=(1.+C(1))/2.
      CTRCHO=C(IMAX)
      DO 80 N=1,NT
      DO 25 I=1,IMAX
25    CO(I)=C(I)
      DO 30 I=2,IMAX1
30    SDC(I)=CO(I-1)-2.*CO(I)+CO(I+1)
      SDC(1)=1.-2.*CO(1)+CO(2)
```

```
      SDC(IMAX)=SDC(IMAX1)
      IF (IOO.LT.0) GO TO 40
      DO 32 I=1,L
32    C(I)=1.0
      DO 35 I=L1,IMAX
      K=I-L
35    C(I)=CO(K)+P*SDC(K)
      CTRACH=C(IMAX)
      GO TO 60
40    CONTINUE
      C(IMAX)=Ctrach0 +N*(CTRACH-Ctrach0)/FNT
      DO 50 I=1,INL
      K=I+L
50    C(I)=CO(K)+P*SDC(K)
      DO 55 I=INL1,IMAX
      F=(I-INL)/FL
55    C(I)=CO(IMAX)+F*(C(IMAX)-CO(IMAX))
60    CONTINUE
80    CONTINUE
      FLUX=D*(C(IMAX)-C(IMAX-1))/DX
      CMOUTH=C(1)
90    FORMAT (10E11.3)
100   CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE INTERP(Y,IMAX1,DX1,IMAX2,DX2)

      DIMENSION Y(201),X1(201),X2(201),Y2(201)

      DO 10 I1=1,IMAX1
10    X1(I1)=(I1-1)*DX1
      DO 20 I2=1,IMAX2
20    X2(I2)=(I2-1)*DX2
      DO 30 I2=1,IMAX2
      I1=X2(I2)/DX1+DX1/2.
      I1=I1+1
      IF (I1.EQ.1) I1=2
      IF (I1.EQ.IMAX1) I1=IMAX1-1
      U=(X2(I2)-X1(I1))/DX1
      D1=Y(I1+1)-Y(I1-1)
      D2=Y(I1+1)-2.*Y(I1)+Y(I1-1)
30    Y2(I2)=Y(I1)+.5*D1*U+.5*D2*U**2
      DO 40 I2=1,IMAX2
40    Y(I2)=Y2(I2)
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE SOLVE(A,B,C,D,NUM,U)
      DIMENSION A(201),B(201),C(201),D(201),U(201)
      A(1)=A(1)/B(1)
      D(1)=D(1)/B(1)
      A(NUM)=0.0
      U(NUM+1)=0.
      DO 10 N=2,NUM
      A(N)=A(N)/(B(N)-C(N)*A(N-1))
10    D(N)=(D(N)+C(N)*D(N-1))/(B(N)-C(N)*A(N-1))
      DO 20 N=1,NUM
      K=NUM+1-N
20.   U(K)=A(K)*U(K+1)+D(K)
      RETURN
      END
C-----------------------------------------------------------------
       SUBROUTINE DERIV(FUN,NUM,H,DFUN)
       DIMENSION FUN(201),DFUN(201)
```

Program CONDIF.FOR

```
      N=NUM-2
      DFUN(1)=(-25.0*FUN(1)/12.0+4.0*FUN(2)-3.0*FUN(3)+4.0*FUN(4)/3.0
     *          -FUN(5)/4.0)/H
      DFUN(2)=(-FUN(1)/4.0-5.0*FUN(2)/6.0+3.0*FUN(3)/2.0-FUN(4)/2.0
     *          +FUN(5)/12.0)/H
      DO 10 N=3,M
 10   DFUN(N+2)=(FUN(N-2)/12.0-2.0*FUN(N-1)/3.0+2.0*FUN(N+1)/3.0
     *              -FUN(N+2)/12.0)/H
      DFUN(NUM-1)=(FUN(NUM)/4.0+5.0*FUN(NUM-1)/6.0-3.0*FUN(NUM-2)/2.0
     *                 +FUN(NUM-3)/2.0-FUN(NUM-4)/12.0)/H

      DFUN(NUM)=(25.0*FUN(NUM)/12.0-4.0*FUN(NUM-1)+3.0*FUN(NUM-2)
     *                -4.0*FUN(NUM-3)/3.0+FUN(NUM-4)/4.0)/H

      RETURN
      END
```

APPENDIX D

Cardiovascular Computer Programs

Appendix D
Cardiovascular Computer Programs

PROGRAM CVMODEL

```
C...
C...  PROGRAM CVMODEL CALLS: (1) SUBROUTINE INITAL TO DEFINE THE ODE
C...  INITAL CONDITIONS, (2) SUBROUTINE RKF45 TO INTEGRATE THE ODES,
C...  AND (3) SUBROUTINE PRINT TO PRINT THE SOLUTION.
C...
C...  THE FOLLOWING CODING IS FOR 500 ODES.  IF MORE ODES ARE TO BE INTE-
C...  GRATED, ALL OF THE 500'S SHOULD BE CHANGED TO THE REQUIRED NUMBER
      IMPLICIT DOUBLE PRECISION (A-H), DOUBLE PRECISION (O-Z)
      INTEGER NI, NO, NEQN, NSTOP, NORUN
      COMMON/T/          T,      NSTOP,      NORUN,     PP,   TIM
     1      /Y/     Y(500)
     2      /F/     F(500)
C...
C...  THE NUMBER OF DIFFERENTIAL EQUATIONS IS IN COMMON/N/ FOR USE IN
C...  SUBROUTINE FCN
      COMMON/N/       NEQN  ! TWO EQUATIONS PER VASCULAR SEGMENT
C...
C...  COMMON AREA TO PROVIDE THE INPUT/OUTPUT UNIT NUMBERS TO OTHER
C...  SUBROUTINES
      COMMON/IO/       NI,        NO
C...
C...  ABSOLUTE DIMENSIONING OF THE ARRAYS REQUIRED BY RKF45
C...
C...    THE USER MUST PROVIDE STORAGE IN HIS CALLING PROGRAM FOR THE ARRAYS
C...    IN THE CALL LIST  -      Y(NEQN) , WORK(3+6*NEQN) , IWORK(5)  ,
C...    DECLARE F IN AN EXTERNAL STATEMENT, SUPPLY SUBROUTINE F(T,Y,YP) AND

      DOUBLE PRECISION YV(500), WORK(3500)
      INTEGER IWORK(5)
C...
C...  EXTERNAL THE DERIVATIVE ROUTINE CALLED BY RKF45
C...
      EXTERNAL FCN
C...
C...  ARRAY FOR THE TITLE (FIRST LINE OF DATA), CHARACTERS  END OF RUNS
      CHARACTER TITLE(20)*4, ENDRUN(3)*4
C...
C...  DEFINE THE CHARACTERS  END OF RUNS
      DATA ENDRUN/'END ','OF R','UNS '/
C...
C...  DEFINE THE INPUT/OUTPUT UNIT NUMBERS
      NI=5
      NO=6
C...
C...  OPEN INPUT AND OUTPUT FILES
      OPEN(NI,FILE='CVDATA.DAT')
      OPEN(NO,FILE='CVOPUT.TXT',BLOCKSIZE=2048)
C...
C...  INITIALIZE THE RUN COUNTER
      NORUN=0
C...
C...  BEGIN A RUN
1     NORUN=NORUN+1
C...
C...  INITIALIZE THE RUN TERMINATION VARIABLE
      NSTOP=0
C...
C...  READ THE FIRST LINE OF DATA
C...
      READ(NI,1000,END=999) (TITLE(I), I = 1, 20)
```

Program CVMODEL.FOR

```
C...
C...  TEST FOR  END OF RUNS  IN THE DATA
C...
      DO 2 I = 1, 3
      IF(TITLE(I) .NE. ENDRUN(I)) GO TO 3
2     CONTINUE
C...
C...  AN END OF RUNS HAS BEEN READ, SO TERMINATE EXECUTION
999   STOP
C...
C...  READ THE SECOND LINE OF DATA
C...
3     READ(NI,*,END=999) TO, TF, TP
C...
C...  READ THE THIRD LINE OF DATA
C...
      READ(NI,*,END=999) NEQN, ERROR
C...
C...  PRINT A DATA SUMMARY
      WRITE(NO,1003)NORUN,(TITLE(I), I = 1, 20),
     1                     TO, TF, TP,
     2                     NEQN, ERROR
      WRITE(*,1003) NORUN, (TITLE(I), I = 1, 20),
     1                     TO, TF, TP,
     2                     NEQN, ERROR
C...
C...  INITIALIZE TIME
      T = TO
C...
C...  SET THE INITIAL CONDITIONS
      CALL INITAL
C...
C...  SET THE INITIAL DERIVATIVES (FOR POSSIBLE PRINTING)
      CALL DERV
C...
C...  PRINT THE INITIAL CONDITIONS
      CALL PRINT(NI, NO)
C...
C...  SET THE INITIAL CONDITIONS FOR SUBROUTINE RKF45
      TV = TO
      DO 5 I = 1, NEQN
      YV(I) = Y(I)
5     CONTINUE
C...
C...  SET THE PARAMETERS FOR SUBROUTINE RKF45
C...
C...   FIRST CALL TO RKF45
C...
      RELERR = ERROR
      ABSERR = ERROR
      IFLAG = 1
      TOUT = TO + TP
C...
C...  CALL SUBROUTINE RKF45 TO START THE SOLUTION FROM THE INITIAL
C...  CONDITION (IFLAG = 1) OR COMPUTE THE SOLUTION TO THE NEXT PRINT
C...  POINT (IFLAG = 2)
C...
4     CALL RKF45(FCN,NEQN,YV,TV,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
C...
C...  PRINT THE SOLUTION AT THE NEXT PRINT POINT
C...
      T=TV
      TOUT = TV + TP
      PRINT *,"Time = ", T
      DO 6 I = 1, NEQN
```

```
         Y(I) = YV(I)
6     CONTINUE
      CALL DERV
      CALL PRINT(NI,NO)
C...
C...  TEST FOR AN ERROR CONDITION
      IF(IFLAG .NE. 2) THEN
C...
C...     PRINT A MESSAGE INDICATING AN ERROR CONDITION
         WRITE(NO,1004) IFLAG
C...
C...     GO ON TO THE NEXT RUN
         GO TO 1
      END IF
C...
C...  CHECK FOR A RUN TERMINATION
      IF(NSTOP .NE. 0) GO TO 1
C...
C...  CHECK FOR THE END OF THE RUN
C...
      IF(TV .LT. (TF - 0.5D0*TP)) GO TO 4
C...
C...  THE CURRENT RUN IS COMPLETE, SO GO ON TO THE NEXT RUN
      GO TO 1
C...
C...  ******************************************************************
C...
C...  FORMATS
C...
1000  FORMAT(20A4)
1001  FORMAT(3E10.0)
1002  FORMAT(I5,20X,E10.0)
1003  FORMAT(1H1,
     1 ' RUN NO. - ',I3,2X,20A4,//,
     2 ' INITIAL T - ',E10.3,//,
     3 '   FINAL T - ',E10.3,//,
     4 '   PRINT T - ',E10.3,//,
     5 ' NUMBER OF DIFFERENTIAL EQUATIONS - ',I5,//,
     6 ' MAXIMUM INTEGRATION ERROR - ',E10.3,//,
     7 1H1)
1004  FORMAT(1H ,//,' IFLAG = ',I3,//,
     1 ' INDICATING AN INTEGRATION ERROR, SO THE CURRENT RUN'     ,/,
     2 ' IS TERMINATED.  PLEASE REFER TO THE DOCUMENTATION FOR'   ,/,
     3 ' SUBROUTINE',//,25X,'RKF45',//,
     4 ' FOR AN EXPLANATION OF THESE ERROR INDICATORS'            )
      END
      SUBROUTINE FCN(TV,YV,YDOT)
C...
C...  SUBROUTINE FCN IS AN INTERFACE ROUTINE BETWEEN SUBROUTINES RKF45
C...  AND DERV
C...
C...  NOTE THAT THE SIZE OF ARRAYS Y AND F IN THE FOLLOWING COMMON AREA
C...  IS ACTUALLY SET BY THE CORRESPONDING COMMON STATEMENT IN MAIN
C...  PROGRAM HEADHIT
      IMPLICIT DOUBLE PRECISION (A-H), DOUBLE PRECISION (O-Z)
      INTEGER NEQN, NSTOP, NORUN
      COMMON/T/          T,      NSTOP,      NORUN
     1      /Y/       Y(500)
     2      /F/       F(500)
C...
C...  THE NUMBER OF DIFFERENTIAL EQUATIONS IS AVAILABLE THROUGH COMMON
C...  /N/
C...
      COMMON/N/       NEQN
C...
```

Program CVMODEL.FOR

```
C...  ABSOLUTE DIMENSION THE DEPENDENT VARIABLE, DERIVATIVE VECTORS
      DOUBLE PRECISION YV(500), YDOT(500)
C...
C...  TRANSFER THE INDEPENDENT VARIABLE, DEPENDENT VARIABLE VECTOR
C...  FOR USE IN SUBROUTINE DERV
C...
      T = TV
      DO 1 I = 1, NEQN
         Y(I) = YV(I)
1     CONTINUE
C...
C...  EVALUATE THE DERIVATIVE VECTOR
C...
      CALL DERV
C...
C...  TRANSFER THE DERIVATIVE VECTOR FOR USE BY SUBROUTINE RKF45
C...
      DO 2 I = 1, NEQN
         YDOT(I) = F(I)
2     CONTINUE
      RETURN
      END
```

Program CVSUBS.FOR

DECK CVSUBS.FOR - SUBROUTINES REQUIRED TO IMPLEMENT A DYNAMIC MODEL OF THE HUMAN

```
C...  CARDIOVASCULAR SYSTEM
C...
C...  LAST REVISION:  1/22/94
C...
      SUBROUTINE INITIAL
C...
C...  THE model described herein parallels the development presented in a paper by
C...  White, RJ, Croston, RC and Fitzjerrell, DG.  Cardiovascular Modelling:  Simulating
C...  the Human Cardiovascular Response to Exercise, Lower Body Negative Pressure, Zero
C...  Gravity and Clinical Conditions.  Adv. Cardiovasc. Phys. (Part I), pp. 195-229 (Karger,
C...  Basel 1983).  It also draws from papers by Jaron, et al who took a similar approach
C..   In particular many of the parameter values for the physical properties of the segments
C...  were taken from:
C...
C...       Jaron, D, Moore, TW, and Bai, J.  Cardiovascular Response to Acceleration Stress:
C...       A Computer Simulation.  Proceedings of the IEEE, Vol 76, No 6, pp. 700-707 (1988).
C...
C...  However, some of the parameters listed in Jaron were clearly in error.  Where new
C...  parameters were required they were derived to yield generally acceptable
C...  flow/pressure/volume and compliance characteristics of the various
C...  cardiovascular subdivisions.  In particular, data from
C...
C...       Burton, Alan, C.  Physiology and Biophysics of the Circulation.  Year
C...       Book Medical Publishers, Inc.  Chicago, IL (1965)
C...
C...    and,
C...
C...       Guyton, A.C.  Textbook of Medical Physiology.  7th Ed.,
C...       W.B. Saunders, Philadelphia, PA. (1985).
C...
C...  The model describes the spatial and temporal variation in the mean
C...  blood pressure along the z-axis of the body.  The model neglects the
C...  non-linear and convective terms in the Navier-Stokes Equation.  The
C...  model also assumes negligible radial flow.  The flow is assumed laminar
C...  except in the ascending an descending aorta where fluid flow resis-
C...  tance multiplied by 33 to account for turbulent pressure losses.
C...
C...  NOTE:  THE SUBSCRIPT NOTATION INDICATES PARTIAL DERIVATIVE WRT THE
C...         SUBSCRIPT  E.G.  Xt ==> THE FIRST PARTIAL OF X WRT TIME
C...         IN THIS MODEL:
C...
C...                    t = Time                       [sec]
C...                    r = Radius of vascular segment  [m]
C...                    l = Length of vascular segment  [m]
C...                 rho0 = density of blood            [kg/m^3]
C...                  mu0 = viscosity of blood          [N-sec/m^2]
C...
C... MODEL FOR ARTERIAL SEGMENTS
C...
C...  The following set of simultaneous equations are solved f    ach
C...  arterial vascular segment.
C...
C...    Pt(t) = 1/C*(Qin(t) - Qout(t)) + R2*(Qtin -Qtcut)
C...    Qt(t) = 1/L*(Pin(t) - Pout(t) + PGz - R*Q(t))
C...    rt(t) = 1/(2*Pi*r*l)*(Qin(t) - Qout(t))
C...
C...    Where,
C...
C...     P = The pressure in the segment                 [Pa]
C...     Q = The segmental volume flow                   [m^3/sec]
C...     C = The capacitance of the segment              [m^3/Pa]
```

Program CVSUBS.FOR

```
C...     L = The inertance of the segment              [kg/m^4] or [Pa-sec^2/m^3]
C...     Ra = The viscous flow resistance in the segment  [m].
C...     PGz = The hydrostatic pressure difference
C...           across the segment because of gravity    [Pa]
C...
C...
C...    And, the following approximations for Ra, La, and Ca are taken from
C...    a paper by
C...
C...       Rideout, et al.  Difference-Differential Equations for Fluid
C...       Flow in Distensible Tubes.  IEEE Transactions on Bio-Medical
C...       Engineering.  Vol BME-14, NO. 3, pp 171-177. Jul 1967.
C...
C...       Ra = 81*mu0*l/(8*pi*r^4)
C...       La =  9*rho0*l^2/(4*VOL)
C...       Ca =  3*r*VOL*l/(2*E*h)
C...
C...    Where,
C...
C...       E = Young's modulus for vessel wall          [Pa]
C...       h = Vessel Wall thickness                    [m].
C...
C...    Finally,
C...
C...       PGz = rho0*Gz*g0*l*cos(theta).
C...
C...    Where,
C...
C...       Gz    = The z-axis "G-level" in units of earth's gravity   [unitless]
C...       g0    = The earth's gravitational acceleration              [m/sec^2]
C...       theta = The angle between the segment and the z-axis       [radians].
C...
C...
C... MODEL FOR VENOUS SEGMENTS
C...
C...    The model for venous segments was adapted from
C...
C...       Snyder, et al.  Computer Simulation Studies of
C...       Venous Circulation.  IEEE Trans Bio-Med Engr Vol BME-16,
C...       NO. 4 pp 325-334. Oct 1969.
C...
C...    The unstressed internal volume of a vascular segment in assumed to be
C...
C...        V = Pi*r^2*l                        [m^3].
C...
C...    When the contained volume, v, is greater than V, the transmural
C...    pressure is assumed to be related to the contained volume by,
C...
C...        dPwall = 1/C*v.
C...
C...    Where C is the vascular compliance as defined above.  For v < V,
C...
C...        dPwall = 1/(20*C)*v.
C...
C...    In a collapsed or partially collapsed vein the flow-pressure
C...    relationship based on an (assumed) elliptical cross-section
C...    and is given by
C...
C...        Qt = 1/Lv*(Pin(t) - Pout(t) + PGz - Rv*Q(t))
C...
C...    Where,
C...
C...       Lv =  9*rho0*l^2/(4*v), and
C...
C...            | 81*mu0*pi^2*l^4*r^2/(8*v^3) for v < V
```

Program CVSUBS.FOR

```
C...        Rv = | or
C...             | 81*mu0*l/(8*pi*r^4)    for v > or = V
C...             | or 81*mu0*l^3/(8*v^2)      ".
C...
C...
C...
C...
C...   PERIPHERAL CAPILLARY BEDS.
C...
C...     Peripheral capillary resistance and capacitance
C...     are modelled as lumped parameter models ("T" circuits)
C...     following the method of Jaron, et al (cited above).
C...
C...
C...
C...                        Ra         Rv
C...             Pa  o--/\/\/\--|--/\/\/\---o Pv
C...                            |
C...                           --- Cperipheral
C...                           -|-
C...                            |
C...                        Pexternal (ie P G-suit)
C...                            |
C...                        Pref (Atmospheric)
C...
C...
C...
C...    PULMONARY CIRCULATION
C...
C...        The pulmonary circulation is modelled as a lumped
C...        parameter model again following Jaron's method.  The
C...        model is a "PI" circuit as shown below.
C...
C...
C...         Prtvent     Rpul
C...      Qvr>---->|---/\/\/\-----| Pleft atrium
C...  venous       |              |
C...  return      --- Cpul        |
C...              -|-             |
C...               |              |
C...          Intra-Thoracic Pressure
C...
C...
C... CARDIAC CIRCULATION/OUTPUT
C...
C...   The chambers of the heart are modelled as variable capacitances
C...   separated by one-way valves.  The pulmonic and aortic valves are
C...   also modelled as one-way valves.  The general method of modelling
C...   The heart and its circulation follows the method of Snyder et al.
C...   Output flow from the left atrium and left ventricle are modelled
C...   as simple half-wave rectified sinusoids whose volume flows are
C...   estimated from pulmonary venous flow.
C...
C...
C... PRESSURE REFERENCE
C...
C...   The pressure reference for the model is located at the tri-cuspid
C...   valve which presumably tracks intrathoracic pressure.
C...
C...
C... MODEL MECHANICS
C...,
C...    For each vessel segment, three coupled non-linear differential
C...    equations must be solved simultaneously.  There are 20 vascular
C...    segments (the pulmonary circuit is segment 1).  There are 10
```

```
C...    capillary bed segments modelled as simple resistance and compliance
C...    circuits which are affected directly by extra-vascular pressure.
C...    The pressures and flows in the various segments are coupled by
C...    their spatial connection.  The following table gives the
C...    approximate anatomical location and the corresponding z-axis
C...    coordinate (measured from the tricuspid valve) for each segment.
C...    The z-axis coordinates were based on a 177 cm tall standing man.
C...
C...                                        Arterial
C...                                        Origin
C... Segment          Anatomical             Z-axis          Peripheral
C... Number           Location               coordinate (cm)   Bed
C...=================================================================
C...    1           Mid-Pulmonary               0               X
C...    2           Ascending Aorta             0
C...    3           Descending Aorta            5
C...    4           Thoracic Aorta/Vena Cava   -8               X
C...    5           Diaphragm/Lower Lung      -15
C...    6           Renal/Hepatic             -22               X
C...    7           Splanchnic                -32               X
C...    8           Buttocks                  -42               X
C...    9           Femoralis                 -50
C...   10           Mid Thigh                 -65               X
C...   11           Knee/Poplateal            - 80
C...   12           Calf                      -100              X
C...   13           Ankle                     -125
C...   14           Foot                      -132              X
C...   15           Aortic Arch                 6
C...   16           Lower Neck                 15
C...   17           Carotid Sinus              25
C...   18           Ophthalmic                 34               X
C...   19           Mid Brain                  37
C...   20           Cerebral                   42               X
C...
C...
C...
C...   Initial Conditions (t = 0)
C...
C...       The initial conditions for pressure, flow, and volume are
C...       set based on a steady-state solution for the model at 1 Gz
C...       for a supine posture.  For other postures, the initial theta's
C...       for the segments must be changed.
C...
C...     Postural and/or Gz changes during a simulation.
C...
C...       This can be most easily accomplished by adding time
C...       varying profiles for Gz and the theta's in SUBROUTINE DERV
C...       which forms the derivatives for CVMODEL.
C...
C...  THE NUMERICAL METHOD OF LINES (W.E. SCHIESSER) IS EMPLOYED TO
C...  INTEGRATE THE COUPLED DIFFERENTIAL EQUATIONS (DES).
C...
C...  ODE COMMON
C...
C...   /Y/ time variables
C...   /F/ time derivatives of variables
C...   /R/ & /I/ real and integer parameters required to define constants and
C...           define the spatial integration grid.
C...
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      PARAMETER (NEQ = 20, NPSEG = 10)
      INTEGER NSTOP, NORUN, IP
      INTEGER*2 ALIN, ALOUT, VLIN, VLOUT, PVS, PIN, POUT
      DOUBLE PRECISION MM2PA, mu0
      COMMON/T/        T,     NSTOP,     NORUN    ! Run Parameters
```

Program CVSUBS.FOR

```
C...
C...   Arrays for segmental variables Pressure (P), Flow(Q), and radii (r)
C...
     1  /Y/     HP(4),        HQ(4),                      ! Heart's Chambers RA=1
     *          AP(NEQ),      AQ(NEQ),     Ar(NEQ),       ! Arterial P, Q, r
     *          VP(NEQ),      VQ(NEQ),     Vr(NEQ),       ! Venous P, Q, r
     *          VOP(3),       VOQ(3),                     ! Venous flows into heart
     *          PP(NEQ),      PQ(NEQ),                    ! Peripheral P, Qin, Qout
C...
C...  Time derivatives of the segmental variables: Pt, Qt, rt
C...
     2  /F/    HPt(4),       HQt(4),                        ! Heart's Chambers RA=1
     *         APt(NEQ),     AQt(NEQ),     Art(NEQ),        ! Arterial Pt, Qt, rt
     *         VPt(NEQ),     VQt(NEQ),     Vrt(NEQ),        ! Venous Pt, Qt, rt
     *         VOPt(3),      VOQt(3),                       ! Venous flows into heart
     *         PPt(NEQ),     PQt(NEQ),                      ! Peripheral Pt, QINt, QOUTt
C...
C... Parameters necessary to form the differential equations
C...
C...    Pi  -  3.14159...
C...    g0  -  9.80665 [m/sec^2]  earth's acceleration of gravity
C...  rho0  -  1050.   [kg/m^3]   density of whole blood (45% Hct)  | Assumed
C...   mu0  -  2.7     [cp]       viscosity of whole blood (45% Hct)| Constant
C...   D2R  -  pi/180  [radians/degree] scale factor
C...
     3  /R/     Pi, g0, rho0, mu0,  D2R, MM2PA, R2,        ! Constants
     *          ZAO(NEQ), ZAT(NEQ), ZVO(NEQ), ZVT(NEQ),    ! Arterial & Venous
     *          THETA(NEQ), HCAP(4), HVOL(4), HVR(4),     ! Orientation angle
     *          ALO(NEQ),  ArU(NEQ),  AE(NEQ), Ah(NEQ),    ! Arterial l, r, E, h
     *          ACAP(NEQ), ARES(NEQ), AINERT(NEQ),         ! Arterial Capacitance, resistance
     *          VLO(NEQ),  VrO(NEQ),  VrU(NEQ),            ! Venous l, r, rUNSTRESSED
     *          VCAPO(NEQ), VCAP(NEQ), VRES(NEQ), VINERT(NEQ), ! Venous Capacitance, resistance, inertance
     *          PRA(NEQ), PRV(NEQ), PCAP(NEQ),             ! Peripheral Ra, Rv, C
     *          PINERT(NEQ), PVOL(NEQ),                    ! Peripheral I, V
     *          PAO(NEQ),  PVO(NEQ),                       ! Initial P conditions
     *          QAO(NEQ),  QVO(NEQ),                       ! Initial Q conditions
     *          AVOL(NEQ),  VVOL(NEQ),                     ! A & V Volumes
     *          PEXT(NEQ),                                 ! Externally applied Pressure
     *          TO, GSTART, GMAX, TBRK1, TBRK2, TMAX, GFIN, ! G-Profile parameters
     *          Gz, ADPG(NEQ), VDPG(NEQ)                   ! Gz & Delta P from G
     4  /I/     IP, NDXPER(NPSEG),                         ! Peripheral Bed Indexes
     *          ALIN(NEQ), ALOUT(NEQ),                     ! Linkage Data arterial
     *          VLIN(NEQ), VLOUT(NEQ),                     ! Venous
     *          PIN(NEQ), POUT(NEQ),                       ! Peripheral
     *          PVS(NEQ),                                  ! Number parallel venous segments
     *          IFOOTSEG, IHEADSEG, IHEARTSEG              ! Foot, Head, & Heart seg nums
C...
C...  Note that artery output flow feeds artery input or peripheral bed
C...  input, artery input comes only from arteries or the heart, venous
C...  output flows to veins or the heart, venous input comes from veins
C...  and/or peripheral beds, peripheral beds are fed only by arteries, and
C...  feed only veins.  The index (i) for an segment refers to its input
C...  flow and pressure.  The output pressure for a segment is
C...  stored in P(i+1) and its output flow is stored at Q(i+1)
C...
C...
C...  Define Some Constants and Parameters
C...
C...  Heart valve resistances
C...
      DATA HVR/ 1.48D+6, 1.48D+6, 2.96D+6, 2.96D+6/
C...
C...  Heart chamber capacitances
C...
      DATA HCAP/ 2.25D-7, 6.55D-7, 1.12D-7, 3.28D-7/
```

Program CVSUBS.FOR

```
C...
C...  Arterial Segment length [meters]
C...
      DATA ALO/ 3.D-2,  3.D-2,  5.D-2,  7.D-2,  5.D-2,
     *         12.D-2, 10.D-2,  8.D-2, 15.D-2, 20.D-2,
     *         15.D-2, 25.D-2,  5.D-2,  2.D-2,  9.D-2,
     *         10.D-2,  9.D-2,  3.D-2,  4.D-2,  1.D-2/
C...
C...  Arterial Segment radii at t = 0  [meters]
C...  These radii are based on flow resistance
C...
      DATA ArU /6.000D-3, 1.500D-2, 1.000D-2, 1.200D-2, 1.000D-2,
     *          6.000D-3, 5.500D-3, 4.000D-3, 3.500D-3, 4.000D-3,
     *          2.800D-3, 4.500D-3, 2.750D-3, 1.000D-3, 6.000D-3,
     *          3.000D-3, 3.000D-3, 3.000D-3, 3.000D-3, 1.500D-3/
C...
C...  These radii are based on capacitance and resistance
C...
C...  Young's modulus for arterial segment walls [Pa]
C...
      DATA AE /  2.5D+5,  5.0D+5,  5.0D+5,  7.0D+5,  7.0D+5,
     *           8.0D+5,  8.0D+5,  8.0D+5,  8.0D+5,  1.0D+6,
     *           1.0D+6,  1.0D+6,  1.0D+6,  1.0D+6,  8.0D+5,
     *           8.0D+5,  8.0D+5,  8.0D+5,  8.0D+5,  8.0D+5/
C...
C...  Wall thickness for arterial segments [meters]
C...
      DATA Ah /  2.D-4,  16.D-4, 16.D-4,  14.D-4, 12.D-4,
     *          12.D-4,  10.D-4, 10.D-4,   8.D-4,  8.D-4,
     *           8.D-4,   6.D-4,  6.D-4,   5.D-4,  5.D-4,
     *           6.D-4,   6.D-4,  6.D-4,   5.D-4,  5.D-4/
C...
C...  Number of parallel venous paths in each segment
C...
      DATA PVS/  4,  1,  1,  2,  4,  4,  4,  4,  2,  2,
     *           2,  2,  4,  4,  2,  2,  4,  4,  4,  4/
C...
C...  Venous segment length [meters]
C...
      DATA VLO/ 10.D-2,  2.D-2,  2.D-2, 10.D-2, 20.D-2,
     *          50.D-2, 40.D-2, 32.D-2, 30.D-2, 40.D-2,
     *          30.D-2, 40.D-2, 20.D-2,  8.D-2, 27.D-2,
     *          20.D-2, 36.D-2, 12.D-2, 16.D-2,  4.D-2/
C...
C...  Initial radii for Venous segment [meters]
C...
      DATA VrU/ 5.0D-3,  5.0D-3,  5.0D-3,  3.0D-3,  3.5D-3,
     *          2.5D-3,  4.0D-3,  4.0D-3,  3.0D-3,  3.0D-3,
     *          3.5D-3,  3.3D-3,  2.5D-3,  1.8D-3,  4.5D-3,
     *          4.0D-3,  4.0D-3,  3.3D-3,  3.0D-3,  2.0D-3/
C...
C...   Venous capacitance [m^3/Pa]
C...
      DATA VCAPO/ 5.0D-8,  5.000D-8,  1.000D-8,  5.000D-8,  5.0D-8,
     *            5.0D-8,  5.000D-8,  4.000D-8,  3.000D-8,  2.5D-8,
     *            8.0D-8,  5.000D-8,  5.000D-8,  5.000D-9,  2.0D-8,
     *            5.0D-8,  8.000D-8,  1.000D-8,  1.000D-8,  3.0D-8/
C...
C...  Peripheral Segment Indices (There are 10 peripheral segments).
C...
      DATA NDXPER / 1, 3, 6, 7, 8, 10, 12, 14, 18, 20/
C...
C...  Peripheral Vascular Capacitance [m^3/Pa]
C...
      DATA PCAP/ 1.13D-7,  0.0D000,  3.75D-8,  0.0D00,  0.0D000,
```

```
     *          7.50D-8,  1.13D-7,  7.50D-8,  0.0000,  3.75D-8,
     *          0.00D00,  3.75D-8,  0.00D00,  3.75D-8, 0.00D00,
     *          0.00D00,  0.00D00,  3.75D-8,  0.00D00, 3.75D-8/
C...
C...  Peripheral Vascular Resistance Arterial side [Pa-sec/m^3]
C...  Assign 9.9D99 to segments with no peripheral bed.
C...
      DATA PRA/ 1.20D+7,  9.99D99,  1.41D+9,  9.99D99,  9.99D99,
     *          1.39D+9,  3.44D+8,  1.37D+9,  9.99D99,  1.34D+9,
     *          9.99D99,  2.18D+9,  9.99D99,  6.34D+9,  9.99D99,
     *          9.99D99,  9.99D99,  1.32D+10, 9.99D99,  6.67D+8/
C...
C...  Peripheral Vascular Resistance Venous side [Pa-sec/m^3]
C...  Assign 9.9D99 to segments with no peripheral bed.
C...
      DATA PRV/ 1.33D+6,  9.99D99, 1.56D+8,  9.99D99,  9.99D99,
     *          1.54D+8,  3.83D+7, 1.52D+8,  9.99D99,  1.49D+8,
     *          9.99D99,  2.43D+8, 9.99D99,  7.04D+8,  9.99D99,
     *          9.99D99,  9.99D99, 1.46D+9,  9.99D99,  7.42D+7/
C...
C...  Initialize extramural pressure vector
C...
      DATA PEXT/ 0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,
     *          0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,
     *          0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,  0.0D+2,
     *          0.0D+2,  0.0D+2,  0.0D+0,  0.0D+0,  0.0D+0/
C...
C...  Set initial flows [m^3/sec]
C...
C...       Arterial flows
C...
      DATA   AQ/ 8.97D-5,  9.00D-5,  9.00D-5,  6.50D-5,  6.50D-5,
     *           6.50D-5,  5.67D-5,  2.33D-5,  1.50D-5,  1.50D-5,
     *           6.67D-6,  6.67D-6,  1.67D-6,  1.67D-6,  1.67D-6,
     *           1.67D-5,  1.67D-5,  1.67D-5,  1.58D-5,  1.58D-5/
C...
C...      Venous flows
C...
      DATA   VQ/ 9.00D-5,  2.00D-5,  7.50D-5,  7.00D-5,  7.00D-5,
     *           6.50D-5,  6.50D-5,  2.33D-5,  1.00D-5,  1.10D-5,
     *           4.70D-6,  7.50D-6,  4.50D-6,  1.67D-6,  1.67D-5,
     *           1.67D-5,  1.67D-5,  1.67D-5,  1.58D-5,  1.58D-5/
C...
C...     Venous output flows into heart
C...
      DATA  VOQ/ 9.0D-5, 1.7D-5, 7.3D-5/
C...
C...       Peripheral flows
C...
      DATA PQ /   8.97D-5,  0.00D00,  8.33D-6,  0.00D00,  0.00D00,
     *            8.33D-6,  3.33D-5,  8.33D-6,  0.00D00,  8.33D-6,
     *            0.00D00,  5.00D-6,  0.00D00,  1.67D-6,  0.00D00,
     *            0.00D00,  0.00D00,  8.33D-7,  0.00D00,  1.58D-5/
C...
C...  Initial Heart Flows
C...
      DATA HQ /   8.97D-5,  8.97D-5,  8.97D-5,  8.97D-5/
C...
C... The following arrays code the linkage between vascular segments
C... Each segment link element contains the index of the next segment
C... in the cardiovascular tree.  For example, ALIN(3) = 4, which
C... means arterial segment 3 feeds arterial segment 4. Segment -1 codes
C... for terminal peripheral beds for arteries and for the heart for veins.
C...
C...  SEGMENT   A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14
```

Program CVSUBS.FOR

```
C...
      DATA ALIN/-1, 3, 4, 5, 6, 7, 8, 9,10, 11, 12, 13, 14, -1,
C...
C...  SEGMENT  A15 A16 A17 A18 A19 A20
C...
     *          16, 17, 18, 19, 20, -1/
C...
C... Venous linkage
C...
C...  SEGMENT    V1  V2 V3 V4 V5 V6 V7 V8  V9 V10 V11 V12 V13 V14
C...
      DATA VLIN/-1, -1, -1, 3, 4, 5, 6, 7,  8,  9, 10, 11, 12, 13,
C...
C...  SEGMENT   V15 V16 V17 V18 V19 V20
C...                                P20
     *            2, 15, 16, 17, 18, 19/
C...
C...  Peripheral Segment Linkage, PIN specifies the input source for the
C...  segment which is only from arteries, POUT specifies the output segment
C...  which is only to veins. "0" means there is no peripheral segment for the
C...  segment number.  Note that the pressure at the inlet of the peripheral
C...  segment is at AP(PIN(i) + 1)
C..
C...  SEGMENT     P1 P2 P3 P4 P5 P6 P7 P8  P9 P10 P11 P12 P13 P14
C...
      DATA PIN  / 0,  0, 3, 0, 0, 6, 7, 8,  0, 10,  0, 12,  0,  0,
C...
C...  SEGMENT   P15 P16 P17 P18 P19 P20
C...
     *            0,  0,  0, 18,  0,  0/
C..
C...  SEGMENT     P1  P2  P3 P4 P5 P6 P7 P8  P9 P10 P11 P12 P13 P14
C...
      DATA  POUT/ 1,  0,   3, 0, 0, 6, 7, 8,  0, 10,  0, 12,  0,  0,
C...
C...  SEGMENT   P15 P16 P17 P18 P19 P20
C...
     *            0,  0,  0, 18,  0,  0/
C...
C...  Z-axis positions relative to tricuspid valve for
C...  the origin of the arterial segment [m]
C...
      DATA ZAO/  0.0D0,   0.0D0,   5.D-2,   0.D-2,  -15.D-2,
     *         -20.D-2, -32.D-2, -42.D-2, -50.D-2,  -65.D-2,
     *         -85.D-2,-100.D-2,-125.D-2,-130.D-2,    5.D-2,
     *          15.D-2,  25.D-2,  34.D-2,  37.D-2,   41.D-2/
C...
C...  Z-axis positions relative to tricuspid valve for
C...  the termination of the arterial segment [m].
C...
      DATA ZAT/  3.D-2,  5.0D-2,   0.D00, -15.D-2,  -20.D-2,
     *         -32.D-2, -42.D-2, -50.D-2, -65.D-2,  -85.D-2,
     *        -100.D-2,-125.D-2,-130.D-2,-132.D-2,   15.D-2,
     *          25.D-2,  34.D-2,  37.D-2,  41.D-2,   42.D-2/
C...
C...  Z-axis positions relative to tricuspid valve for
C...  the origin of the venous segment [m].
C...
      DATA ZVO/ 3.0D-2,1.5D-2, -1.5D-2, -15.D-2, -20.D-2,
     *        -32.D-2, -42.D-2, -50.D-2, -65.D-2, -85.D-2,
     *       -100.D-2,-125.D-2,-130.D-2,-132.D-2,  15.D-2,
     *         25.D-2,  34.D-2,  37.D-2,  41.D-2,  42.D-2/
C...
C...  Z-axis positions relative to tricuspid valve for
C...  the termination of the venous segment [m].
```

Program CVSUBS.FOR

```
C...
      DATA ZVT/ 0.0D0,  0.0D00,  0.0D00, -1.5D-2,  -15.D-2,
     *         -20.D-2, -32.D-2, -42.D-2, -50.D-2,  -65.D-2,
     *         -85.D-2,-100.D-2,-125.D-2,-130.D-2,   1.5D-2,
     *          15.D-2,  25.D-2,  34.D-2,  37.D-2,   41.D-2/
C...
C... Initial orientations (degrees) of z-axis projection of vascular
C... segments.  Arterial and venous assumed to be at the same orientation.
C...
      DATA THETA/ 20.D0, 20.D0, 20.D0, 20.D0,  20.D0,  ! Partially reclined
     *            20.D0, 20.D0, 20.D0, 90.D0,  90.D0,  ! Seated position
     *            90.D0, 110.D0, 110.D0, 90.D0, 20.D0, ! 20 deg SBA
     *            0.D0, 0.D0, 0.D0, 0.D0,  0.D0/
C...
C... Define some physical constants
C...
      Pi      = DACOS(-1.0D0)           ! pi
      g0      = 9.80665D0               ! m/sec^2 - earth's gravity
      rho0    = 1050.0D0                ! kg/m^3  - density of whole blood
      mu0     = 2.70D-3                 ! N-sec/m^2 - fluid viscosity
      D2R     = Pi/180.D0               ! scale from degrees to radians
      MM2PA   = 1.01325D5/760.D0        ! scale from mmHg to Pascals
      Gz      = 4.D0                    ! Initial Gz
      R2      = 0.002D0                 ! Jaron's wall energy term
C...
C...  INITIAL CONDITIONS (T = 0)
C...
C...  The pressures are set assuming a prone posture ie transverse g0.
C...  The initial pressures are assigned by linearly interpolating
C...  pressures from the foot to the heart and from the heart to the
C...  cerebral segment. The mean arterial pressure is assumed
C...  to be 100 mmHg at the heart and 95 mm Hg at both the foot and
C...  cerebral segments.  The venous pressure is assumed to be 2 mmHg
C...  at the heart and 5 mmHg at both the foot and cerebral segments.
C...
      PAFOOT    =  95.D0*MM2PA
      PAHEAD    =  95.D0*MM2PA
      IFOOTSEG  = 14
      IHEADSEG  = 20
      IHEARTSEG =  1
      PRATRM =    PEXT(1)         ! Right Atrial Pressure
      HP(1)  = PRATRM + MM2PA     ! Inlet Pressure to Right Atrium
      HDP1   =    2.D0*MM2PA      ! Increase in right atrial pressure
      PRVENT =  HP(1) + HDP1      ! Inlet Pressure to Right Ventricular
      HP(2)  = PRVENT
      HDP2   = 11.D0*MM2PA        ! Increase in right ventricular pressure
      PLATRM =  4.D0*MM2PA       ! Left Atrial Pressure
      HP(3)  = PLATRM             ! Left Atrial Inlet Pressure
      HDP3   = 4.D0*MM2PA         !
      PLVENT = HP(3) + HDP3       ! Left Ventricular Pressure
      HP(4)  = PLVENT             ! Inlet Pressure to LV
      HDP4   = 92.D0*MM2PA        ! Increase in LV Pressure
C...
C...  G-Profile parameters
C...
      T0 = 0.D0
      GSTART = 1.D0
      GMAX = 3.D0
      TMAX = 400.D0
      TBRK1 = 10.D0
      TBRK2 = 300.D0
      GFIN =  3.D0
      i = 1
      DO WHILE (i .LE. NEQ)
         THETA(i) = THETA(i)*D2R
```

```
         i = i + 1
      END DO
      i = 1
      DO WHILE (i .LE. NEQ)
         SCALEF = 1.D0
         IF( i .LE. 5 .OR. i .EQ. 15 ) SCALEF = 33.D0
         ARES(i) = 81.D0*mu0*ALO(i)/(8.D0*Pi*ArU(i)**4)*SCALEF
         DPVS = DBLE(PVS(i))
         Vr(i) = DMAX1(Vr(i), 1.33D0*VrU(i))
         VRES(i) = 81.D0*mu0*(VLO(i)/DPVS)/(8.D0*Pi*Vr(i)**4)
         i = i + 1
      END DO
      AP(1) = HP(2) + HDP2 + 2.D0*MM2PA
      AP(2) = HP(4) + HDP4 + 2.D0*MM2PA
      i = 3
      DO WHILE ( i .LE. IFOOTSEG)
         AP(i) = AP(i-1) - AQ(i-1)*ARES(i-1)
     *           + rho0*Gz*g0*(ZAO(i)-ZAT(i))*DCOS(THETA(i))
         i = i + 1
      END DO
      AP(15) = AP(3)
      i = 16
      DO WHILE ( i .LE. IHEADSEG)
         AP(i) = AP(i-1) - AQ(i-1)*ARES(i-1)
     *           + rho0*Gz*g0*(ZAO(i)-ZAT(i))*DCOS(THETA(i))
         i = i + 1
      END DO
C...
C...  Set the initial radii and arterial capacitance
C...
C
      i = 1
      DO WHILE (i .LE. NEQ)
         dPwall  = AP(i)
         deltaR  = 7.5D-1*dPwall*ArU(i)**2/(AE(i)*Ah(i))
         Ar(i)   = ArU(i) + deltaR
         ACAP(i) = 3.D0*Pi*Ar(i)**3*ALO(i)/(2.D0*AE(i)*Ah(i))
         deltaR  = dPwall*ACAP(i)/(2.D0*Pi*Ar(i)*ALO(i))
         Ar(i)   = ArU(i) + deltaR
         AVOL(i) = Pi*Ar(i)**2*ALO(i)
         i = i + 1
      END DO
C...
C...  Initial Venous Pressures
C...
      VOP(2) = HP(1)
      VOP(3) = HP(1)
      VP(1) = HP(3)  + VQ(1)*VRES(1)
     *      - rho0*Gz*g0*(ZVO(1) - ZVT(1))*DCOS(THETA(1))
      VP(2) = VOP(2) + VQ(2)*VRES(2)
     *      - rho0*Gz*g0*(ZVO(2) - ZVT(2))*DCOS(THETA(2))
      VP(3) = VOP(3) + VQ(3)*VRES(3)
     *      - rho0*Gz*g0*(ZVO(3) - ZVT(3))*DCOS(THETA(3))
      i = 4
      DO WHILE ( i .LE. IFOOTSEG)
         VP(i) = VP(i-1) + VQ(i)*VRES(i)
     *           - rho0*Gz*g0*(ZVO(i) - ZVT(i))*DCOS(THETA(i))
         i = i + 1
      END DO
      VP(15) = VP(2) + VQ(15)*VRES(15)
     *               - rho0*Gz*g0*(ZVO(15) - ZVT(15))*DCOS(THETA(15))
      i = 16
      DO WHILE ( i .LE. IHEADSEG)
         VP(i) = VP(i-1) + VQ(i)*VRES(i)
     *           - rho0*Gz*g0*(ZVO(i) - ZVT(i))*DCOS(THETA(i))
```

```
         i = i + 1
      END DO
C...
C...  Compute the initial resistances
C...
      i = 1
      DO WHILE (i .LE. NEQ)
         VCAP(i) = VCAPO(i)
         dPwall  = VP(i) - PEXT(i)
         deltaR  = dPwall*VCAP(i)/(2.DO*Pi*VrU(i)*VLO(i))
         Vr(i)   = VrU(i) + deltaR
         deltaR  = dPwall*VCAP(i)/(2.DO*Pi*Vr(i)*VLO(i))
         Vr(i)   = VrU(i) + deltaR
         VVOL(i) = Pi*Vr(i)**2*VLO(i)
         i = i + 1
      END DO

C...
C...   Initial Heart flows
C...
      HQ(1) = 9.D-5
      HQ(2) = 9.D-5
      HQ(3) = 9.D-5
      HQ(4) = 9.D-5

C...
C...  Set initial Peripheral Pressures
C...
      PP(1) =  PQ(1)*PRA(1) + VQ(1)*PRV(1) + VP(1)
      i = 3
      DO WHILE ( i .LE. NEQ)
         IF( PIN(i) .GT. 0 )  THEN
             PP(i) = AP(i+1)
         END IF
         i = i + 1
      END DO
      PP(14) =  PQ(14)*PRA(14) + VQ(14)*PRV(14) + VP(14)
      PP(20) =  PQ(20)*PRA(20) + VQ(20)*PRV(20) + VP(20)
C...
C...   Call DERV to set initial derivatives -- loop to stabilize derivatives
C...
      i = 1
      DO WHILE (i .LE. 100)
         CALL DERV
         i = i + 1
      END DO
      IP=0
      RETURN
      END
C...
      SUBROUTINE DERV
C...
C...  DERV CALCULATES THE TIME DERIVATIVES TO BE INTEGRATED BY RKF45
C...
C...  ODE COMMON
C...
C...   /Y/ time variables
C...   /F/ time derivatives of variables
C...   /S/ spatial derivatives of variables
C...   /R/ & /I/ real and integer parameters required to define constants and
C...           define the spatial integration grid.
C...
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      PARAMETER (NEQ = 20, NPSEG = 10)
      INTEGER NSTOP, NORUN, IP
```

Program CVSUBS.FOR

```
      INTEGER*2 ALIN, ALOUT, VLIN, VLOUT, PVS, PIN, POUT
      DOUBLE PRECISION MM2PA, mu0
      COMMON/T/         T,     NSTOP,     NORUN    ! Run Parameters
C...
C...   Arrays for segmental variables Pressure (P), Flow(Q), and radii (r)
C...
     1  /Y/    HP(4),     HQ(4),                   ! Heart's Chambers RA=1
     *      AP(NEQ),   AQ(NEQ),   Ar(NEQ),         ! Arterial P, Q, r
     *      VP(NEQ),   VQ(NEQ),   Vr(NEQ),         ! Venous P, Q, r
     *      VOP(3),     VOQ(3),                    ! Venous flows into heart
     *      PP(NEQ),   PQ(NEQ),                    ! Peripheral P, Qin, Qout
C...
C...  Time derivatives of the segmental variables: Pt, Qt, rt
C...
     2  /F/   HPt(4),    HQt(4),                   ! Heart's Chambers RA=1
     *     APt(NEQ),  AQt(NEQ),   Art(NEQ),        ! Arterial Pt, Qt, rt
     *     VPt(NEQ),  VQt(NEQ),   Vrt(NEQ),        ! Venous Pt, Qt, rt
     *     VOPt(3),    VOQt(3),                   ! Venous flows into heart
     *     PPt(NEQ),  PQt(NEQ),                    ! Peripheral Pt, QINt, QOUTt
C...
C... Parameters necessary to form the differential equations
C...
C...    Pi -  3.14159...
C...    g0 -  9.80665 [m/sec^2]  earth's acceleration of gravity
C...  rho0 -  1050.   [kg/m^3]   density of whole blood (45% Hct)  | Assumed
C...   mu0 -  2.7     [cp]       viscosity of whole blood (45% Hct)| Constant
C...   D2R -  pi/180  [radians/degree] scale factor
C...
     3  /R/  Pi, g0, rho0, mu0,  D2R, MM2PA, R2,      ! Constants
     *       ZAO(NEQ), ZAT(NEQ), ZVO(NEQ), ZVT(NEQ),   ! Arterial & Venous
     *       THETA(NEQ), HCAP(4), HVOL(4), HVR(4),                   ! Orientation angle
     *       ALO(NEQ), ArU(NEQ),  AE(NEQ), Ah(NEQ),    ! Arterial l, r, E, h
     *       ACAP(NEQ), ARES(NEQ), AINERT(NEQ),        ! Arterial Capacitance, resistance
     *       VLO(NEQ),  VrO(NEQ),  VrU(NEQ),           ! Venous l, r, rUNSTRESSED
     *       VCAPO(NEQ), VCAP(NEQ), VRES(NEQ), VINERT(NEQ), ! Venous Capacitance, resistance, inertance
     *       PRA(NEQ), PRV(NEQ), PCAP(NEQ),            ! Peripheral Ra, Rv, C
     *       PINERT(NEQ), PVOL(NEQ),                   ! Peripheral I, V
     *       PAO(NEQ),  PVO(NEQ),                      ! Initial P conditions
     *       QAO(NEQ),  QVO(NEQ),                      ! Initial Q conditions
     *       AVOL(NEQ),  VVOL(NEQ),                    ! A & V Volumes
     *       PEXT(NEQ),                                ! Externally applied Pressure
     *       TO, GSTART, GMAX, TBRK1, TBRK2, TMAX, GFIN, ! G-Profile parameters
     *       Gz, ADPG(NEQ), VDPG(NEQ)                  ! Gz & Delta P from G
     4  /I/  IP, NDXPER(NPSEG),                        ! Peripheral Bed Indexes
     *       ALIN(NEQ), ALOUT(NEQ),                    ! Linkage Data arterial
     *       VLIN(NEQ), VLOUT(NEQ),                    ! Venous
     *       PIN(NEQ), POUT(NEQ),                      ! Peripheral
     *       PVS(NEQ),                                 ! Number parallel venous segments
     *       IFOOTSEG, IHEADSEG, IHEARTSEG             ! Foot, Head, & Heart seg nums
      DIMENSION PQOUT(NEQ)
C...
C...   Right Heart
C...
      PRATRM =   PEXT(1)         ! Right Atrial Pressure
      HP(1)  = PRATRM + MM2PA    ! Inlet Pressure to Right Atrium
      HDP1   =    1.D0*MM2PA     ! Increase in right atrial pressure
      PRVENT =  HP(1) + HDP1     ! Inlet Pressure to Right Ventricular
      HP(2)  = PRVENT
      HDP2   = 11.D0*MM2PA       ! Increase in right ventricular pressure
      PINTHO =  -1.0D0 *MM2PA    ! Intra-thoracic Pressure                     ! Extramural
pressure for the pulmonary bed
C...
C...  Left Heart
C...
      PLATRM =   4.D0*MM2PA      ! Left Atrial Pressure
```

Program CVSUBS.FOR

```
      HP(3)  = PLATRM           ! Left Atrial Inlet Pressure
      HDP3   = 4.D0*MM2PA       !
      PLVENT = HP(3) + HDP3     ! Left Ventricular Pressure
      HP(4)  = PLVENT           ! Inlet Pressure to LV
      HDP4   = 92.D0*MM2PA      ! Increase in LV Pressure
C...
C...   Ensure pressures are not less than external pressures
C...
      i = 1
      DO WHILE ( i .LE. NEQ )
         IF (VQ(i) .LT. 0.D0) THEN
             VQ(i)  = 0.D0 ! Venous valves
         END IF
         i = i + 1
      END DO
      i = 1
      DO WHILE (i .LE. 3)
         VOQ(i) = DMAX1(VOQ(i), 0.D0)
         i = i + 1
      END DO
      AQ(1) = DMAX1(AQ(1), 0.D0) ! No reverse flow into the heart
      AQ(2) = DMAX1(AQ(2), 0.D0) ! from arteries or out of the heart
      HQ(1) = DMAX1(HQ(1), 0.D0) ! into veins
      HQ(3) = DMAX1(HQ(3), 0.D0)
C...
C...  Set the Heart level G level
C...
C     CALL TRAPG(T0,GSTART,GMAX,TMAX,TBRK1,TBRK2,GFIN,T,Gz)
C...
C...  Set peripheral inlet pressures to the appropriate
C...  arterial outlet pressures
C...
      i = 3
      DO WHILE ( i .LE. NEQ)
         IF( PIN(i) .GT. 0 )  THEN
             PP(i) = AP(i+1)
             PPt(i) = APt(i+1)
         END IF
         i = i + 1
      END DO
C...
C...  Compute the resistance, capacitance and inertance for the arterial segments
C...
      AP(15) = AP(3)
      i = 1
      DO WHILE (i .LE. NEQ)
         IF ( AP(i) .LE. PEXT(i) ) THEN
            Ar(i) = ArU(i)   ! If arterial pressure is below external
         END IF      ! pressure set radius to minimum (unstressed).
C...
C...   rationalize pressure, radius, and capacitance.
C...
         dPwall = AP(i) - PEXT(i)
         IF(dPwall .LE. 0.D0) Ar(i) = ArU(i)
         ACAP(i) = 3.D0*Pi*Ar(i)**3*ALO(i)/(2.D0*AE(i)*Ah(i))
         deltaR  = dPwall*ACAP(i)/(2.D0*Pi*Ar(i)*ALO(i))
C         Ar(i)   = ArU(i) + deltaR
         AVOL(i) = Pi*Ar(i)**2*ALO(i)
         SCALEF = 1.D0
         IF( i .LE. 5 .OR. i .EQ. 15 ) SCALEF = 33.D0
         ARES(i)   = 81.D0*mu0*ALO(i)/(8.D0*Pi*Ar(i)**4)*SCALEF
         AINERT(i) =  9.D0*rho0*ALO(i)**2/(4 D0*AVOL(i))
         ADPG(i)   = rho0*Gz*g0*(ZAO(i) - ZAT(i))*DCOS(THETA(i))
         i = i + 1
      END DO
```

```
C...
C...  Venous resistance, capacitance and inertance for the venous segments
C...
      i = 1
      DO WHILE (i .LE. NEQ)
         dPwall = DMAX1(VP(i) - PEXT(i),0.D0)
         IF(dPwall .LE. 0.D0) Vr(i) = VrU(i)
         Vrmax = 1.333*VrU(i)
         Vr(i)  = DMAX1(Vr(i), VrU(i))
         Vr(i)  = DMIN1(Vr(i), Vrmax)
         DPVS = DBLE(PVS(i))
         VVOL(i) = Pi*Vr(i)**2*VLO(i)
         VRES(i) = 81.D0*mu0*(VLO(i)/DPVS)/(8.D0*Pi*Vr(i)**4)
         VINERT(i) = 9.D0*rho0*VLO(i)**2/(4.D0*VVOL(i))
         VDPG(i) = rho0*Gz*g0*(ZVO(i) - ZVT(i))*DCOS(THETA(i))
         i = i + 1
      END DO
      CAPL = 1.D-2 ! Capillary Length
      i = 1
      DO WHILE( i .LE. NPSEG)
         j = NDXPER(i)
         dPwall = DMAX1(PP(j) - PEXT(j), MM2PA)  ! Min pressure 1 mmHg
         PVOL(j) = PCAP(j) * dPwall
         i = i + 1
      END DO
C...
C...    Define the differential equations describing pressure, flow, and
C...    radius
C...
C...    Pt(t) = 1/C*(Qin(t) - Qout(t))  + R2/C*(Qtin(t)- Qtout(t))
C...    Qt(t) = 1/L*(Pin(t) - Pout(t) + PGz - Pext - R*Q(t))
C...    rt(t) = 1/(2*Pi*r*l)*(Qin(t) - Qout(t))
C...
C...
C...      Infow to the right atrium HQ(1) = VOQ(2) + VOQ(3)
C...
        IF (VQ(2) .LE. 0.D0) VP(2) = HP(1)
        VOQt(2) = 1.D0/VINERT(2)*(VP(2) - HP(1) + VDPG(2)   ! Outflow from the Superior V.C.
     *            - VRES(2)*VOQ(2))
        IF (VQ(3) .LE. 0.D0) VP(3) = HP(1)
        VOQt(3) = 1.D0/VINERT(3)*(VP(3) - HP(1) + VDPG(3)   ! Outflow from the Inferior V.C.
     *            - VRES(3)*VOQ(3))
        HQt(1) = VOQt(2) + VOQt(3)
        HQ(1) = VOQ(2) + VOQ(3)

C...
C...   Right Atrium   [Heart Segment 1]
C...
      HQ2 = HQ(1) - HQ(2)
      HQ2t = HQt(1) - HQt(2)
      HR2 = 2.D-5/HCAP(1)
      HPt(1) = HQ2t*HR2 + HQ2/HCAP(1)
      HPt(1) = 0.D0
      HPt(2) = 0.D0
      HQ(2) = (HP(1) + HDP1 + MM2PA - HP(2))/HVR(1)
      HQt(2) = (HPt(1) - HPt(2))/HVR(1)
      HQt(2) = HQt(1)
      HQ(2) =  HQ(1)
C...
C...  Right Ventricle   [Heart Segment 2]
C...
      HQ2 = HQ(2) - AQ(1)
      HQ2t = HQt(2) - AQt(1)
      HR2 = 2.D-5/HCAP(2)
      HPt(2) = HQ2t*HR2 + HQ2/HCAP(2)
```

Program CVSUBS.FOR

```
      HPt(2) = 0.D0
C...
C...   Pulmonary Circulation
C...
C...  Pulmonary Artery    [Arterial Segment 1]
C...
      AQ2  = AQ(1) - PQ(1)
      AQ2t = AQt(1) - PQt(1)
      AR2 = R2/ACAP(1)
      APt(1) = AQ2t*AR2 + AQ2/ACAP(1)
      APt(1) = 0.D0
      Art(1) =  1.D0/(2.D0*Pi*Ar(1)*ALO(1))*AQ2
      Art(1) =  0.D0
      AQ(1) = (HP(2) + HDP2 + 2.D0*MM2PA - AP(1))/HVR(2)
      AQt(1) = (HPt(2) - APt(1))/HVR(2)
      AQ(1) = HQ(2)
C...
C...  Pulmonary Capillary Bed   [Peripheral Segment 1]
C...
      PQt(1) = 1.D0/AINERT(1)*(AP(1) - PP(1) + ADPG(1)
     *          - ARES(1)*PQ(1))
      PQ2 = PQ(1) - VQ(1)
      PPt(1) = PRA(1)*PQt(1) + PQ2/PCAP(1)
C...
C...  Pulmonary Veins
C...
      VQt(1) = (PPt(1) - VPt(1) - PQt(1)*PRA(1))/PRV(1)
      VQ2  = VQ(1)  - HQ(3)
      VQ2t = VQt(1) - HQt(3)
      VR2 = R2/VCAP(1)
      VPt(1) = VQ2t*VR2 + VQ2/VCAP(1)
      Vrt(1) =  1.D0/(2.D0*Pi*Vr(1)*VLO(1))*VQ2
C      Vrt(1) =  0.D0
C...
C...  Left Atrium   [Heart Segment 3]
C...
      HQt(3) = 1.D0/VINERT(1)*(VP(1) - HP(3) + VDPG(1)
     *          - HQ(3)*VRES(1))
      VOQt(1) = HQt(3)
      HQ2 = HQ(3) - HQ(4)
      HQ2t = HQt(3) - HQt(4)
      HR2 = 2.D-5/HCAP(3)
      HPt(3) = HQ2t*HR2 + HQ2/HCAP(3)
      HPt(3) = 0.D0
C...
C...  Left Ventricle   [Heart Segment 4]
C...
      HQ(4) = (HP(3) + HDP3 + 2.D0*MM2PA - HP(4))/HVR(3)
      HQ(4) = HQ(3)
      HQt(4) = (HPt(3) - HPt(4))/HVR(3)
      HQt(4) = HQt(3)
      HQ2 = HQ(4) - AQ(2)
      HQ2t = HQt(4) - AQt(2)
      HR2 = 2.D-5/ACAP(4)
      HPt(4) = HQ2t*HR2 + HQ2/HCAP(4)
      HPt(4) = 0.D0
C...
C...  Ascending Aortic artery [Arterial Segment 2]
C...
C...  AQ(2) = (HP(4) + HDP4 + 2.D0*MM2PA - AP(2))/HVR(4)
      AQt(2) = (HPt(4) - APt(2))/HVR(4)
      AQ2 = AQ(2) - AQ(3) - AQ(15)
      AQ2t = AQt(2) - AQt(3) - AQt(15)
      AR2 = R2/ACAP(2)
      APt(2) = AQ2t*AR2 + AQ2/ACAP(2)
```

```
      APt(2) = 0.D0
      Art(2) = 1.D0/(2.D0*Pi*Ar(2)*ALO(2))*AQ2
      Art(2) = 0.D0
C...
C...  Descending Aorta   [Arterial Segment 3]
C...
      AQt(3) =  1.D0/AINERT(2)*(AP(2) - AP(3)
     *          + ADPG(2) - ARES(2)*(AQ(3) + AQ(15))) - AQt(15)
      AQ2  =  AQ(3) - AQ(4)  - PQ(3)
      AQ2t = AQt(3) - AQt(4) - PQt(3)
      AR2 = R2/ACAP(3)
      APt(3) = AQ2t*AR2 + AQ2/ACAP(3)
      AQt(4) = 1.D0/AINERT(3)*(AP(3) - AP(4) + ADPG(3)
     *          - ARES(3)*(AQ(4) + PQ(3))) - PQt(3)
      Art(3) =  1.D0/(2.D0*Pi*Ar(3)*ALO(3))*AQ2
      Art(3) = 0.D0
C...
C...  Form the derivatives for the other arterial segments below the heart.
C...
C...  Thoracic and Cardiac   [Arterial Segment 4]
C...
      AQ2 = AQ(4) - AQ(5)
      AQ2t = AQt(4) - AQt(5)
      AR2 = R2/ACAP(4)
      APt(4) =  1.D0/ACAP(4)*AQ2 + AR2*AQ2t
      AQt(5) =  1.D0/AINERT(4)*(AP(4) - AP(5)
     *        + ADPG(4) - ARES(4)*AQ(5))
      Art(4) =  1.D0/(2.D0*Pi*Ar(4)*ALO(4))*AQ2
      Art(4) = 0.D0
C...
C...  Diaphragm  [Arterial Segment 5]
C...
      AQ2 = AQ(5) - AQ(6)
      AQ2t = AQt(5) - AQt(6)
      AR2 = R2/ACAP(5)
      APt(5) =  AR2*AQ2t + AQ2/ACAP(5)
      AQt(6) =  1.D0/AINERT(5)*(AP(5) - AP(6)
     *        + ADPG(5) - ARES(5)*AQ(6))
      Art(5) =  1.D0/(2.D0*Pi*Ar(5)*ALO(5))*AQ2
      Art(5) = 0.D0
C...
C...  Renal - Hepatic   [Arterial Segment 6]
C...
      AQ2 = AQ(6) - AQ(7) - PQ(6)
      AQ2t = AQt(6) - AQt(7) - PQt(6)
      AR2 = R2/ACAP(6)
      APt(6) =  AR2*AQ2t + AQ2/ACAP(6)
      AQt(7) =  1.D0/AINERT(6)*(AP(6) - AP(7)
     *        + ADPG(6) - ARES(6)*(AQ(7) + PQ(6))) - PQt(6)
      Art(6) =  1.D0/(2.D0*Pi*Ar(6)*ALO(6))*AQ2
      Art(6) = 0.D0
C...
C...  Splanchnic  [Arterial Segment 7]
C...
      AQ2 = AQ(7) - AQ(8) - PQ(7)
      AQ2t = AQt(7) - AQt(8) - PQt(7)
      AR2 = R2/ACAP(7)
      APt(7) =  AR2*AQ2t + AQ2/ACAP(7)
      AQt(8) =  1.D0/AINERT(7)*(AP(7) - AP(8)
     *        + ADPG(7) - ARES(7)*(AQ(8) + PQ(7))) - PQt(7)
      Art(7) =  1.D0/(2.D0*Pi*Ar(7)*ALO(7))*AQ2
      Art(7) = 0.D0
C...
C...  Buttocks   [Arterial Segment 8]
C...
```

```
      AQ2 = AQ(8) - AQ(9) - PQ(8)
      AQ2t = AQt(8) - AQt(9) - PQt(8)
      AR2 = R2/ACAP(8)
      APt(8) =  1.D0/ACAP(8)*AQ2 + AR2*AQ2t
      AQt(9) =  1.D0/AINERT(8)*(AP(8) - AP(9)
     *          + ADPG(8) - ARES(8)*(AQ(9) + PQ(8))) - PQt(8)
      Art(8) =  1.D0/(2.D0*Pi*Ar(8)*ALO(8))*AQ2
      Art(8) = 0.D0
C...
C...  Femoralis   [Arterial Segment 9]
C...
      AQ2 = AQ(9) - AQ(10)
      AQ2t = AQt(9) - AQt(10)
      AR2 = R2/ACAP(9)
      APt(9) =  AR2*AQ2t + AQ2/ACAP(9)
      AQt(10) =  1.D0/AINERT(9)*(AP(9) - AP(10)
     *          + ADPG(9) - ARES(9)*AQ(10))
      Art(9) =  1.D0/(2.D0*Pi*Ar(9)*ALO(9))*AQ2
      Art(9) = 0.D0
C...
C...  Thigh   [Arterial Segment 10]
C...
      AQ2 = AQ(10) - AQ(11) - PQ(10)
      AQ2t = AQt(10) - AQt(11) - PQt(10)
      AR2 = R2/ACAP(10)
      APt(10) =  AR2*AQ2t + AQ2/ACAP(10)
      AQt(11) =  1.D0/AINERT(10)*(AP(10) - AP(11)
     *          + ADPG(10) - ARES(10)*(AQ(11)+PQ(10))) - PQt(10)
      Art(10) =  1.D0/(2.D0*Pi*Ar(10)*ALO(10))*AQ2
      Art(10) = 0.D0
C...
C...  Knee   [Arterial Segment 11]
C...
      AQ2 = AQ(11) - AQ(12)
      AQ2t = AQt(11) - AQt(12)
      AR2 = R2/ACAP(11)
      APt(11) =  AQ2/ACAP(11) +  AR2*AQ2t
      AQt(12) =  1.D0/AINERT(11)*(AP(11) - AP(12)
     *              +  ADPG(11) - ARES(11)*AQ(12))
      Art(11) =  1.D0/(2.D0*Pi*Ar(11)*ALO(11))*AQ2
      Art(11) =  0.D0
C...
C...  Calf   [Arterial Segment 12]
C...
      AQ2 = AQ(12) - AQ(13) - PQ(12)
      AQ2t = AQt(12) - AQt(13) - PQt(12)
      AR2 = R2/ACAP(12)
      APt(12) =  AR2*AQ2t + AQ2/ACAP(12)
      AQt(13) =  1.D0/AINERT(12)*(AP(12) - AP(13)
     *          + ADPG(12) - ARES(12)*(AQ(13)+PQ(12))) - PQt(12)
      Art(12) =  1.D0/(2.D0*Pi*Ar(12)*ALO(12))*AQ2
      Art(12) = 0.D0
C...
C...  Ankle   [Arterial Segment 13]
C...
      AQ2 = AQ(13) - AQ(14)
      AQ2t = AQt(13) - AQt(14)
      AR2 = R2/ACAP(13)
      APt(13) =  AQ2t*AR2 + AQ2/ACAP(13)
      AQt(14) =  1.D0/AINERT(13)*(AP(13) - AP(14)
     *          + ADPG(13) - ARES(13)*AQ(14))
      Art(13) =  1.D0/(2.D0*Pi*Ar(13)*ALO(13))*AQ2
      Art(13) = 0.D0
C...
C...   Foot   [Arterial Segment 14]
```

```
C...
      AQ2 = AQ(14) - PQ(14)
      AQ2t = AQt(14) - PQt(14)
      AR2 = R2/ACAP(14)
      APt(14) = AR2*AQ2t + AQ2/ACAP(14)
      PQt(14) = 1.D0/AINERT(14)*(AP(14) - PP(14) + ADPG(14)
     *          - ARES(14)*PQ(14))
      Art(14) =  1.D0/(2.D0*Pi*Ar(14)*ALO(14))*AQ2
      Art(14) = 0.D0
C...
C...   Peripheral Bed in Foot   [Peripheral Segment 14]
C...
      VQ(14) = (PP(14) - VP(14) - PQ(14)*PRA(14))/PRV(14)
      PQ2 = PQ(14) - VQ(14)
      PPt(14) = PQt(14)*PRA(14) + PQ2/PCAP(14)
C...
C...  Venous Drainage from Foot   [Venous Segment 14]
C...
      VQt(14) = (PPt(14) - VPt(14) - PQt(14)*PRA(14))/PRV(14)
      VQ2 = VQ(14) - VQ(13)
      VQ2t = VQt(14) - VQt(13)
      VR2 = R2/VCAP(14)
      VPt(14) = VQ2t*VR2 + VQ2/VCAP(14)
      VQt(13) = 1.D0/VINERT(14)*(VP(14) - VP(13) + VDPG(14)
     *          - VRES(14)*VQ(13))
      Vrt(14) =  1.D0/(2.D0*Pi*Vr(14)*VLO(14))*VQ2
C...
C...  Ankle  [Venous Segment 13]
C...
      PQOUT(12) = (PP(12) - VP(12) - PRA(12)*PQ(12))/PRV(12)
      PQOUTt = (PPt(12) - VPt(12) - PRA(12)*PQt(12))/PRV(12)
      VQ2 =  VQ(13) - (VQ(12) - PQOUT(12))
      VQ2t = VQt(13) - (VQt(12) - PQOUTt)
      VR2  = R2/VCAP(13)
      VPt(13) = VQ2t*VR2 + VQ2/VCAP(13)
      VQt(12) = 1.D0/VINERT(13)*(VP(13) - VP(12) + VDPG(13)
     *          - VRES(13)*(VQ(12) - PQOUT(12))) + PQOUTt
      Vrt(13) =  1.D0/(2.D0*Pi*Vr(13)*VLO(13))*VQ2
C...
C...  Calf   [Venous Segment 12]
C...
      VQ2 = VQ(12) - VQ(11)
      VQ2t = VQt(12) - VQt(11)
      VR2 = R2/VCAP(12)
      VPt(12) = VQ2t*VR2 + VQ2/VCAP(12)
      VQt(11) = 1.D0/VINERT(12)*(VP(12) - VP(11) + VDPG(12)
     *          - VRES(12)*VQ(11))
      Vrt(12) =  1.D0/(2.D0*Pi*Vr(12)*VLO(12))*VQ2
C...
C...  Knee   [Venous Segment 11]
C...
      PQOUT(10) = (PP(10) - VP(10) - PRA(10)*PQ(10))/PRV(10)
      PQOUTt = (PPt(10) - VPt(10) - PRA(10)*PQt(10))/PRV(10)
      VQ2 =  VQ(11) - (VQ(10) - PQOUT(10))
      VQ2t = VQt(11) - (VQt(10) - PQOUTt)
      VR2  = R2/VCAP(11)
      VPt(11) = VQ2t*VR2 + VQ2/VCAP(11)
      VQt(10) = 1.D0/VINERT(11)*(VP(11) - VP(10) + VDPG(11)
     *          - VRES(11)*(VQ(10) - PQOUT(10))) + PQOUTt
      Vrt(11) =  1.D0/(2.D0*Pi*Vr(11)*VLO(11))*VQ2
C...
C...  Thigh   [Venous Segment 10]
C...
      VQ2 = VQ(10) - VQ(9)
      VQ2t = VQt(10) - VQt(9)
```

```
      VR2 = R2/VCAP(10)
      VPt(10) = VQ2t*VR2 + VQ2/VCAP(10)
      VQt(9) = 1.D0/VINERT(10)*(VP(10) - VP(9) + VDPG(10)
     *           - VRES(10)*VQ(9))
      Vrt(10) =  1.D0/(2.D0*Pi*Vr(10)*VL0(10))*VQ2
C...
C...   Femoralis  [Venous Segment 9]
C...
      PQOUT(8) = (PP(8) - VP(8) - PRA(8)*PQ(8))/PRV(8)
      PQOUTt = (PPt(8) - VPt(8) - PRA(8)*PQt(8))/PRV(8)
      VQ2 =  VQ(9) - (VQ(8) - PQOUT(8))
      VQ2t = VQt(9) - (VQt(8) - PQOUTt)
      VR2  = R2/VCAP(9)
      VPt(9) = VQ2t*VR2 + VQ2/VCAP(9)
      VQt(8) = 1.D0/VINERT(9)*(VP(9) - VP(8) + VDPG(9)
     *           - VRES(9)*(VQ(8) - PQOUT(8))) + PQOUTt
      Vrt(9) =  1.D0/(2.D0*Pi*Vr(9)*VL0(9))*VQ2
C...
C...   Buttocks   [Venous Segment 8]
C...
      PQOUT(7) = (PP(7) - VP(7) - PRA(7)*PQ(7))/PRV(7)
      PQOUTt = (PPt(7) - VPt(7) - PRA(7)*PQt(7))/PRV(7)
      VQ2 =  VQ(8) - (VQ(7) - PQOUT(7))
      VQ2t = VQt(8) - (VQt(7) - PQOUTt)
      VR2  = R2/VCAP(8)
      VPt(8) = VQ2t*VR2 + VQ2/VCAP(8)
      VQt(7) = 1.D0/VINERT(8)*(VP(8) - VP(7) + VDPG(8)
     *           - VRES(8)*(VQ(7) - PQOUT(7))) + PQOUTt
      Vrt(8) =  1.D0/(2.D0*Pi*Vr(8)*VL0(8))*VQ2
C...
C...   Splanchnic   [Venous Segment 7]
C...
      PQOUT(6) = (PP(6) - VP(6) - PRA(6)*PQ(6))/PRV(6)
      PQOUTt = (PPt(6) - VPt(6) - PRA(6)*PQt(6))/PRV(6)
      VQ2 =  VQ(7) - (VQ(6) - PQOUT(6))
      VQ2t = VQt(7) - (VQt(6) - PQOUTt)
      VR2  = R2/VCAP(7)
      VPt(7) = VQ2t*VR2 + VQ2/VCAP(7)
      VQt(6) = 1.D0/VINERT(7)*(VP(7) - VP(6) + VDPG(7)
     *           - VRES(7)*(VQ(6) - PQOUT(6))) + PQOUTt
      Vrt(7) =  1.D0/(2.D0*Pi*Vr(7)*VL0(7))*VQ2
C...
C...   Renal - Hepatic   [Venous Segment 6]
C...
      VQ2 = VQ(6) - VQ(5)
      VQ2t = VQt(6) - VQt(5)
      VR2 = R2/VCAP(6)
      VPt(6) = VQ2t*VR2 + VQ2/VCAP(6)
      VQt(5) = 1.D0/VINERT(6)*(VP(6) - VP(5) + VDPG(6)
     *           - VRES(6)*VQ(5))
      Vrt(6) =  1.D0/(2.D0*Pi*Vr(6)*VL0(6))*VQ2
C...
C...   Diaphragm   [Venous Segment 5]
C...
      VQ2 = VQ(5) - VQ(4)
      VQ2t = VQt(5) - VQt(4)
      VR2 = R2/VCAP(5)
      VPt(5) = VQ2t*VR2 + VQ2/VCAP(5)
      VQt(4) = 1.D0/VINERT(5)*(VP(5) - VP(4) + VDPG(5)
     *           - VRES(5)*VQ(4))
      Vrt(5) =  1.D0/(2.D0*Pi*Vr(5)*VL0(5))*VQ2
C...
C...   Thoracic Circulation [Venous Segment 4]
C...
        PQOUT(3) = (PP(3) - VP(3) - PRA(3)*PQ(3))/PRV(3)
```

```
      PQOUTt = (PPt(3) - VPt(3) - PRA(3)*PQt(3))/PRV(3)
      VQ2 =  VQ(4) - (VQ(3) - PQOUT(3))
      VQ2t = VQt(4) - (VQt(3) - PQOUTt)
      VR2  = R2/VCAP(4)
      VPt(4) = VQ2t*VR2 + VQ2/VCAP(4)
      VQt(3) = 1.D0/VINERT(4)*(VP(4) - VP(3) + VDPG(4)
     *           - VRES(4)*(VQ(3) - PQOUT(3))) + PQOUTt
      Vrt(4) =  1.D0/(2.D0*Pi*Vr(4)*VLO(4))*VQ2
C...
C...   Inferior Vena Cava   [Venous Segment 3]
C...
      VQ2  = VQ(3)  - VOQ(3)
      VQ2t = VQt(3) - VOQt(3)
      VR2  = R2/VCAP(3)
      VPt(3) = VQ2t*VR2 + VQ2/VCAP(3)
      Vrt(3) = 1.D0/(2.D0*Pi*Vr(3)*VLO(3))*VQ2
C...
C...
C...  Above the heart
C...
C...   Subclavian - Upper Thorax   [Arterial Segment 15]
C...
      AP(15) = AP(3)
      AQ2 = AQ(15) - AQ(16)
      AQ2t = AQt(15) - AQt(16)
      AR2 = R2/ACAP(15)
      APt(15) = AQ2t*AR2 + AQ2/ACAP(15)
      AQt(15) =  1.D0/AINERT(2)*(AP(2) - AP(15)
     *          +  ADPG(2) - ARES(2)*(AQ(3) + AQ(15))) - AQt(3)
      AQt(16) = 1.D0/AINERT(15)*(AP(15) - AP(16) + ADPG(15)
     *          - ARES(15)*AQ(16))
      Art(15) =  1.D0/(2.D0*Pi*Ar(15)*ALO(15))*AQ2
      Art(15) = 0.D0
C...
C...  Lower Neck   [Arterial Segment 16]
C...
      AQ2 = AQ(16) - AQ(17)
      AQ2t = AQt(16) - AQt(17)
      AR2 = R2/ACAP(16)
      APt(16) = AQ2t*AR2 + AQ2/ACAP(16)
      AQt(17) =  1.D0/AINERT(16)*(AP(16) - AP(17)
     *       +  ADPG(16) - ARES(16)*AQ(17))
      Art(16) =  1.D0/(2.D0*Pi*Ar(16)*ALO(16))*AQ2
       Art(16) = 0.D0
C...
C...     Upper Neck (Carotid sinus)   [Arterial Segment 17]
C...
      AQ2 =  AQ(17) - AQ(18)
      AQ2t = AQt(17) - AQt(18)
      AR2 = R2/ACAP(17)
      APt(17) = AQ2t*AR2 + AQ2/ACAP(17)
      AQt(18) = 1.D0/AINERT(17)*(AP(17) - AP(18)
     *          + ADPG(17) - ARES(17)*AQ(18))
      Art(17) =  1.D0/(2.D0*Pi*Ar(17)*ALO(17))*AQ2
      Art(17) = 0.D0
C...
C...   Opthalmic   [Arterial Segment 18]
C...
      AQ2 = AQ(18) - (AQ(19) + PQ(18))
      AQ2t = AQt(18) - (AQt(19) + PQt(18))
      AR2 = R2/ACAP(18)
      APt(18) = AQ2t*AR2 + AQ2/ACAP(18)
      AQt(19) = 1.D0/AINERT(18)*(AP(18) - AP(19) + ADPG(18)
     *           - ARES(18)*(AQ(19) + PQ(18))) - PQt(18)
      Art(18) =  1.D0/(2.D0*Pi*Ar(18)*ALO(18))*AQ2
```

Program CVSUBS.FOR

```
      Art(18) =  0.D0
C...
C...   Midbrain   [Arterial Segment 19]
C...
      AQ2 = AQ(19) - AQ(20)
      AQ2t = AQt(19) - AQt(20)
      AR2 = R2/ACAP(19)
      APt(19) = AQ2t*AR2 + AQ2/ACAP(19)
      AQt(20) = 1.D0/AINERT(19)*(AP(19) - AP(20) + ADPG(19)
     *           - ARES(19)*AQ(20))
      Art(19) =  1.D0/(2.D0*Pi*Ar(19)*ALO(19))*AQ2
      Art(19) = 0.D0
C...
C...   Cerebral  [Arterial Segment 20]
C...
      AQ2 = AQ(20) - PQ(20)
      AQ2t = AQt(20) - PQt(20)
      AR2 = R2/ACAP(20)
      APt(20) = AQ2t*AR2 + AQ2/ACAP(20)
      Art(20) =  1.D0/(2.D0*Pi*Ar(20)*ALO(20))*AQ2
      Art(20) = 0.D0
C...
C...  Cerebral bed   [Peripheral Segment 20]
C...
      PQt(20) = 1.D0/AINERT(20)*(AP(20) - PP(20) + ADPG(20)
     *           - ARES(20)*PQ(20))
      VQ(20) = DMAX1((PP(20) - VP(20) - PQ(20)*PRA(20))/PRV(20),0.D0)
      PQ2 = PQ(20) - VQ(20)
      PPt(20) = PQt(20)*PRA(20) + PQ2/PCAP(20)
C...
C...  Venous Drainage from Brain  [Venous Segment 20]
C...
      VQt(20) = (PPt(20) - VPt(20) - PQt(20)*PRA(20))/PRV(20)
      VQ2 = VQ(20) - VQ(19)
      VQ2t = VQt(20) - VQt(19)
      VR2 = R2/VCAP(20)
      VPt(20) = VQ2t*VR2 + VQ2/VCAP(20)
      VQt(19) = 1.D0/VINERT(20)*(VP(20) - VP(19) + VDPG(20)
     *           - VRES(20)*VQ(19))
      Vrt(20) =  1.D0/(2.D0*Pi*Vr(20)*VLO(20))*VQ2
C...
C...  Midbrain   [Venous Segment 19]
C...
      PQOUT(18) = (PP(18) - VP(18) - PRA(18)*PQ(18))/PRV(18)
      PQOUTt = (PPt(18) - VPt(18) - PRA(18)*PQt(18))/PRV(18)
      VQ2 =  VQ(19) - (VQ(18) - PQOUT(18))
      VQ2t = VQt(19) - (VQt(18) - PQOUTt)
      VR2  = R2/VCAP(19)
      VPt(19) = VQ2t*VR2 + VQ2/VCAP(19)
      VQt(18) = 1.D0/VINERT(19)*(VP(19) - VP(18) + VDPG(19)
     *           - VRES(19)*(VQ(18) - PQOUT(18))) + PQOUTt
      Vrt(19) =  1.D0/(2.D0*Pi*Vr(19)*VLO(19))*VQ2
C...
C...   Opthalmic   [Venous Segment 18]
C...
      VQ2 =  VQ(18) - VQ(17)
      VQ2t = VQt(18) - VQt(17)
      VR2 = R2/VCAP(18)
      VPt(18) = VQ2t*VR2 + VQ2/VCAP(18)
      VQt(17) = 1.D0/VINERT(18)*(VP(18) - VP(17) + VDPG(18)
     *           - VRES(18)*VQ(17))
      Vrt(18) =  1.D0/(2.D0*Pi*Vr(18)*VLO(18))*VQ2
C...
C...   Upper Neck   [Venous Segment 17]
C...
```

```
      VQ2 =  VQ(17) - VQ(16)
      VQ2t = VQt(17) - VQt(16)
      VR2 = R2/VCAP(17)
      VPt(17) = VQ2t*VR2 + VQ2/VCAP(17)
      VQt(16) = 1.D0/VINERT(17)*(VP(17) - VP(16) + VDPG(17)
     *          - VRES(17)*VQ(16))
      Vrt(17) =  1.D0/(2.D0*Pi*Vr(17)*VLO(17))*VQ2
C...
C...   Lower Neck (Juglar)   [Venous Segment 16]
C...
      VQ2 =  VQ(16) - VQ(15)
      VQ2t = VQt(16) - VQt(15)
      VR2 = R2/VCAP(16)
      VPt(16) = VQ2t*VR2 + VQ2/VCAP(16)
      VQt(15) = 1.D0/VINERT(16)*(VP(16) - VP(15) + VDPG(16)
     *          - VRES(16)*VQ(15))
      Vrt(16) =  1.D0/(2.D0*Pi*Vr(16)*VLO(16))*VQ2
C...
C...   Subclavian   [Venous Segment 15]
C...
      VQ2 =  VQ(15) - VQ(2)
      VQ2t = VQt(15) - VQt(2)
      VR2 = R2/VCAP(15)
      VPt(15) = VQ2t*VR2 + VQ2/VCAP(15)
      VQt(2) = 1.D0/VINERT(15)*(VP(15) - VP(2) + VDPG(15)
     *          - VRES(15)*VQ(2))
      Vrt(15) =  1.D0/(2.D0*Pi*Vr(15)*VLO(15))*VQ2
C...
C...  Superior Vena Cava  [Venous Segment 2]
C...
        VQ2 =  VQ(2)  - VOQ(2)
        VQ2t = VQt(2) - VOQt(2)
        VR2 = R2/VCAP(2)
        VPt(2) = VQ2t*VR2 + VQ2/VCAP(2)
        Vrt(2) =  1.D0/(2.D0*Pi*Vr(2)*VLO(2))*VQ2
C...
C...  Form the derivatives for the peripheral beds
C...
C...   Opthalmic   [Peripheral Segment 18]
C...
      PQ2  = PQ(18) - PQOUT(18)
      PQt(18) = PPt(18)/PRA(18) - PQ2/(PRA(18)*PCAP(18))
C...
C...  Thorax and Coronaries [Peripheral Segment 3]
C...
      PQ2  = PQ(3)  - PQOUT(3)
      PQt(3) =  PPt(3)/PRA(3) - PQ2/(PRA(3)*PCAP(3))
C...
C...   Remainder of Peripheral Segments
C...
      i = 6
      DO WHILE(i .LE. IFOOTSEG - 1)
         IF ( POUT(i) .GT. 0 ) THEN
            PQ2  = PQ(i) - PQOUT(i)
            PQt(i) =  PPt(i)/PRA(i) - PQ2/(PRA(i)*PCAP(i))
         END IF
         i = i + 1
      END DO
      RETURN
      END
      SUBROUTINE PRINT(NI,NO6, NO7, NO8)
C...
C...  ODE COMMON
C...
C...   /Y/ time variables
```

Program CVSUBS.FOR

```
C...   /F/ time derivatives of variables
C...   /S/ spatial derivatives of variables
C...   /R/ & /I/ real and integer parameters required to define constants and
C...            define the spatial integration grid.
C...
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      PARAMETER (NEQ = 20, NPSEG = 10)
      INTEGER NSTOP, NORUN, IP
      INTEGER*2 ALIN, ALOUT, VLIN, VLOUT, PVS, PIN, POUT
      DOUBLE PRECISION MM2PA, mu0, PPbed(NEQ)
      COMMON/T/         T,      NSTOP,      NORUN    ! Run Parameters
C...
C...   Arrays for segmental variables Pressure (P), Flow(Q), and radii (r)
C...
     1  /Y/    HP(4),     HQ(4),                     ! Heart's Chambers RA=1
     *        AP(NEQ),   AQ(NEQ),   Ar(NEQ),       ! Arterial P, Q, r
     *        VP(NEQ),   VQ(NEQ),   Vr(NEQ),       ! Venous P, Q, r
     *        VOP(3),     VOQ(3),                    ! Venous flows into heart
     *        PP(NEQ),   PQ(NEQ),                    ! Peripheral P, Qin, Qout
C...
C...  Time derivatives of the segmental variables: Pt, Qt, rt
C...
     2  /F/  HPt(4),     HQt(4),                          ! Heart's Chambers RA=1
     *      APt(NEQ),  AQt(NEQ),   Art(NEQ),          ! Arterial Pt, Qt, rt
     *      VPt(NEQ),  VQt(NEQ),   Vrt(NEQ),          ! Venous Pt, Qt, rt
     *        VOPt(3),   VOQt(3),                         ! Venous flows into heart
     *      PPt(NEQ),  PQt(NEQ),                          ! Peripheral Pt, QINt, QOUTt
C...
C... Parameters necessary to form the differential equations
C...
C...    Pi -  3.14159...
C...    g0 -  9.80665 [m/sec^2]  earth's acceleration of gravity
C...  rho0 -  1050.   [kg/m^3]   density of whole blood (45% Hct)  | Assumed
C...   mu0 -  2.7     [cp]       viscosity of whole blood (45% Hct)| Constant
C...   D2R -  pi/180  [radians/degree] scale factor
C...
     3  /R/  Pi, g0, rho0, mu0,  D2R, MM2PA, R2,          ! Constants
     *        ZAO(NEQ), ZAT(NEQ), ZVO(NEQ), ZVT(NEQ),    ! Arterial & Venous
     *        THETA(NEQ), HCAP(4), HVOL(4), HVR(4),                            ! Orientation angle
     *        ALO(NEQ), ArU(NEQ),  AE(NEQ), Ah(NEQ),      ! Arterial l, r, E, h
     *        ACAP(NEQ), ARES(NEQ), AINERT(NEQ),           ! Arterial Capacitance, resistance
     *        VLO(NEQ),  VrO(NEQ),  VrU(NEQ),              ! Venous l, r, rUNSTRESSED
     *        VCAPO(NEQ), VCAP(NEQ), VRES(NEQ), VINERT(NEQ), ! Venous Capacitance, resistance, inertance
     *        PRA(NEQ),  PRV(NEQ), PCAP(NEQ),              ! Peripheral Ra, Rv, C
     *        PINERT(NEQ), PVOL(NEQ),                      ! Peripheral I, V
     *        PAO(NEQ),  PVO(NEQ),                         ! Initial P conditions
     *        QAO(NEQ),  QVO(NEQ),                         ! Initial Q conditions
     *        AVOL(NEQ), VVOL(NEQ),                        ! A & V Volumes
     *        PEXT(NEQ),                                   ! Externally applied Pressure
     *        TO, GSTART, GMAX, TBRK1, TBRK2, TMAX, GFIN, ! G-Profile parameters
     *        Gz, ADPG(NEQ), VDPG(NEQ)                     ! Gz & Delta P from G
     4  /I/  IP, NDXPER(NPSEG),                           ! Peripheral Bed Indexes
     *        ALIN(NEQ), ALOUT(NEQ),                       ! Linkage Data arterial
     *        VLIN(NEQ), VLOUT(NEQ),                       ! Venous
     *        PIN(NEQ), POUT(NEQ),                       ! Peripheral
     *        PVS(NEQ),                                    ! Number parallel venous segments
     *        IFOOTSEG, IHEADSEG, IHEARTSEG                ! Foot, Head, & Heart seg nums
C...
C...
C...  PRINT A HEADING FOR THE NUMERICAL SOLUTION
      IP=IP+1
      IF(IP.EQ.1)WRITE(NO6,100)
      IF(IP.EQ.1)WRITE(NO7,100)
      IF(IP.EQ.1)WRITE(NO8,100)
      IF(IP.EQ.1)WRITE(*,1)
```

Program CVSUBS.FOR

```
C      WRITE(NO,2)
C      WRITE(NO,2) T
   1  FORMAT(2X,'T = ',F12.4,' sec'  )
C...
C...  PRINT THE SOLUTION
C...
      HQ(1) = VOQ(2) + VOQ(3)
      WRITE( *,22) T, Gz, THETA(1),(VP(k), k=1,NEQ), (VQ(k), k=1,NEQ)
      WRITE( *,22)  T, Gz, THETA(1),(HP(k), k=1,4), (HQ(k), k=1,4)
      WRITE(NO6,2) T, Gz, THETA(1),(AP(k), k=1,NEQ), (AQ(k), k=1,NEQ)
      WRITE(NO7,2) T, Gz, THETA(1),(VP(k), k=1,NEQ), (VQ(k), k=1,NEQ)
      i = 1
      DO WHILE (i .LE. NEQ)
        PPbed(i) = PP(i) - PQ(i)*PRA(i)
        i = i + 1
      END DO
      WRITE(NO8,2) T, Gz, THETA(1),(PPbed(k), k=1,NEQ),(PQ(k), k=1,NEQ)
   2  FORMAT(3F8.4,60(E12.4,1X))
  22  FORMAT(1X,3F8.4, /,12(5(E12.4,1X),1X,/))
 100  FORMAT(2X,' Time',3X,'     Gz     ','    Theta    ','Op Art Press',
     *' Op Art Flow',' Op Vn Press',' Op Vn Flow','Op PBed Pr  ')
      RETURN
      END
```

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Abboud, S; Bruderman, I; Sadeh, D. Frequency and Time Domain Analysis of Airflow Breath Patterns in Patients with Chronic Obstructive Airway Disease. Comput Biomed Res. 19: 266-273; 1986.

2. Ackles, KN; Porlier, JAG; Holness, DE; Wright, GR; Lambert, JN; McArthur, WJ. Protection Against the Physiological Effects of Positive Pressure Breathing. Aviat Space Environ Med. 49(6): 753-758; 1978.

3. Air Standard Agreement 61/21. Air Standardization and Coordinating Committee. Washington, DC. 1982 August 18.

4. Albery, WB. The Effect of Sustained Acceleration and Noise on Workload in Human Operations. Aviat Space Environ Med. 60: 943-948; 1989.

5. Anthonisen, NR; Danson, J; Robertson, PC; Ross, WRD. Airway Closure as a Function of Age. Respir Physiol. 8: 58-65; 1969/70.

6. Anthonisen, NR; Robertson, PC; Ross, WRD. Gravity-dependent sequential emptying of lung regions. J Appl Physiol. 28(5): 589-595; 1970.

7. Anthonisen, NR; Peress, LJ; Siegler, DIM; Dhingra, S. Lung Volume, Volume History and the Distribution of Inhales Boluses. Respir Physiol. 33: 279-288; 1978.

8. Arborelius Jr, M; Dahlback, GO; Data, PG. Cardiac output and gas exchange during heavy exercise with a positive pressure respiratory protective apparatus. Scand J Work Environ Health. 9: 471-477; 1983.

9. Arieli, R; Van Liew, HD. Post-Inspiratory Mixing in the Lung. Prog Resp Res. 16: 93-95; 1981.

10. Asmussen, E; Nielsen, M. Studies on the Regulation of Respiration in Heavy Work. Acta Physiol Scand. 12: 171-187; 1946.

11. Astrand, PO; Rodahl, K. Energy Cost of Various Activities. IN: Textbook of Work Physiology. New York: McGraw-Hill; 1970; Chapter 13: 433-450.

12. Astrand, PO; Saltin, B. Oxygen uptake during the first minutes of heavy muscular exercise. J Appl Physiol. 16(6): 971-976; 1961.

13. Avanzolini, G; Barbini, P. Comments on "Estimating Respiratory Mechanical Parameters in Parallel Compartment Models". IEEE Trans Biomed Eng. BME-29(12): 772-774; 1982.

14. Avanzolini, G; Barbini, P; Cappello, A; Cevenini, G. Parameter Estimates in Ventilatory Mechanics Models During Induced RDS. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 1987: 1804-1806.

15. Avanzolini, G; Barbini, P. Sensitivity Analysis for an Improved Estimation of Respiratory Mechanics Parameters. J Biomed Eng. 6(July): 189-194; 1984.

16. Avanzolini, G; Barbini, P. A Versatile Identification Method Applied to Analysis of Respiratory Mechanics. IEEE Trans Biomed Eng. BMD-31(7): 520-526; 1984.

17. Avula, XJR; Ostreicher, HL. Mathematical Model of the Cardiovascular System Under Acceleration Stress. Aviat Space Environ Med. 49(1): 279-286; 1978.

18. Baconnier, PF; Eberhard, A; Grimbert, FA. Theoretical Analysis of Occlusion Techniques for Measuring Pulmonary Capillary Pressure. J Appl Physiol. 73(4): 1351-1359; 1992 October.

19. Bailey, HR; Gogarty, WB. Numerical and experimental results on the dispersion of a solute in a fluid in laminar flow through a tube. Proc R Soc (London) A: Mathematical & Physical Sciences. 269: 352-367; 1962.

20. Bailliart, O; Normand, H; Marotte, H; Bonnin, P; Vargas, E. Distribution of Common Carotid Blood Flow, Measured by Doppler, in Man at High Altitude. Aviat Space Environ Med. 61(12): 1102-1106; 1990.

21. Bake, B; Wood, L; Murphy, B; Macklem, PT; Milic-Emili, J. Effect of inspiratory flow rate on regional distribution of inspired gas. J Appl Physiol. 37(1): 8-17; 1974.

22. Baker, LG; Ultman, JS; Rhoades, RA. Simultaneous Gas Flow and Diffusion in a Symmetric Airway System: A Mathematical Model. Respir Physiol. 21: 119-138; 1974.

23. Balldin, U. Decompression Sickness and Nitrogen Elimination in Man: Influence of Immersion, Body Position and Ambient Temperature with special reference to hemodynamics. Thesis, Laboratory of Aviation and Naval Physiology, Institute of Physiology, University of Lund, Sweden. 1-51; 1973.

24. Balldin, U; Liner, M. Preventive Effect of a Vasodilator on the Occurrence of Decompression Sickness in Rabbits. Aviat Space Environ Med. 49(6): 759-762; 1978.

25. Balldin, UI; Lundgren, CEG; Lundvall, J; Mellander, S. Changes in the Elimination of 133Xenon from the Anterior Tibial Muscle in Man Induced by Immersion in Water and by Shifts in Body Position. Aerospace Med. 42(5): 489-493; 1971.

26. Balldin, UI; Lundgren, CEG. Effects of Immersion with the Head Above Water on Tissue Nitrogen Elimination in Man. Aerospace Med. 43(10): 1101-1108; 1972.

27. Balldin, UI. Effects of Ambient Temperature and Body Position on Tissue Nitrogen Elimination in Man. Aerospace Med. 44(4): 365-370; 1973.

28. Balldin, UI. The effects of body position and a vasodilator on xenon133 elimination from human subcutaneous fat. Undersea Biomed Res. 3(4): 379-385; 1976.

29. Balldin, UI. Explosive Decompression of Subjects up to a 20,000-m Altitude Using a Two-Pressure Flying Suit. Aviat Space Environ Med. 49(4): 599-602; 1978.

30. Balldin, UI; Siegborn, J. G-Endurance During Heat Stress and Balanced Pressure Breathing. Aviat Space Environ Med. 63(March): 177-180; 1992.

31. Balldin, UI; Wranne, B. Hemodynamic effects of extreme positive pressure breathing using a two-pressure flying suit. Aviat Space Environ Med. 51(9): 851-855; 1980 September.

32. Balldin, UI. Influence of Preoxygenation on Intracardial Gas Bubbles and Decompression Sickness During Flying After Diving. Proceedings of the 5th Annual Scientific Meeting of the European Undersea Biomedical Society; July 5-6, 1979; Bergen, Norway. Norway: G. Grimstad; 1980: 130-140.

33. Balldin, UI; Borgstrom, P. Intracardial Bubbles During Decompression to Altitude in Relation to Decompression Sickness in Man. Aviat Space Environ Med. 47(2): 113-116; 1976.

34. Balldin, UI; Borgstrom, P. Intracardial Gas Bubbles at Altitude After Negative Pressure Breathing. Aviat Space Environ Med. 48(11): 1007-1011; 1977.

35. Balldin, UI. The Preventive Effect of Denitrogenation during Warm Water Immersion on Decompression Sickness in Man. Sartryck ur Forsvarsmedicin. 9(3): 1-7; 1973.

36. Balldin, UI; Liner, MH. Xe133 elimination from human fat during negative- and positive-pressure breathing. Undersea Biomed Res. 3(2): 163-169; 1976.

37. Barer, AS; Breslav, IS; Isayev, GG; YaA, S. The effects of increased respiratory resistance on human work capacity. Kosm Biol Aviakosm Med (USSR Space Life Sciences Digest; Issue 23; NASA Contractor Report 3922(27); Contract NASW-4292; August 1989); 23(2): 4-11; 1989.

38. Bartsch, P; Haeberli, A; Hauser, K; Gubser, A; Straub, PW. Fibrinogenolysis in the Absence of Fibrin Formation in Severe Hypobaric Hypoxia. Aviat Space Environ Med. 59: 428-432; 1988.

39. Bashoff, MA; Ingram Jr, RH; Schilder, DP. Effect of expiratory flow rate on the nitrogen concentration vs. volume relationship. J Appl Physiol. 23(6): 895-901; 1967.

40. Bates, JHT; Brown, K; Kochi, T. Identifying a Model of Respiratory Mechanics Using the Interrupter Technique. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 1802-1803; 1987.

41. Bennett, FM; Fordyce, WE. Control of Breathing During Exercise: A Simulation Study. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 2056-2057; 1987.

42. Berend, N; Skoog, C; Thurlbeck, WM. Collateral ventilation in excised human lungs. J Appl Physiol. 50(5): 927-930; 1981.

43. Berend, N; Thurlbeck, WM. Exponential analysis of pressure-volume relationship in excised human lungs. J Appl Physiol. 52(4): 838-844; 1982.

44. Bergel, DH. The Dynamic Elastic Properties of the Arterial Wall. J Physiol. 156: 458-469; 1961.

45. Bergman, NA; Waltemath, CL. A comparison of some methods for measuring total respiratory resistance. J Appl Physiol. 36(1): 131-134; 1974 January.

46. Bert, JL; Pinder, KL. Pulmonary Microvascular Exchange: An Analog Computer Simulation. Microvasc Res. 27(1): 51-70; 1984 January.

47. Bhansali, PV; Irvin, CG; Dempsey, JA; Bush, R; Webster, JG. Human pulmonary resistance: effect of frequency and gas physical properties. J Appl Physiol. 47(1): 161-168; 1979.

48. Bidani, A. Analysis of Abnormalities of Capillary CO2 Exchange In Vivo. J Appl Physiol. 70(4): 1686-1699; 1991 April.

49. Billings, CE; Ernsting, J. Protection Afforded By Phased Dilution Oxygen Equipment Following Rapid Decompression: Performance Aspects. Aerospace Med. 45(2): 132-134; 1974.

50. Bjurstedt, H; Rosenhamer, G; Lindborg, B; Hesser, CM. Respiratory and circulatory responses to sustained positive-pressure breathing and exercise in man. Acta Physiol Scand. 105: 204-214; 1979.

51. Blide, RW; Kerr, HD; Spicer Jr, WS. Measurement of upper and lower airway resistance and conductance in man. J Appl Physiol. 19(6): 1059-1069; 1964.

52. Bomar, J. Personal Notes. USAF School of Aerospace Medicine. 1981.

53. Bomar, JB. "New Developments in Altitude Protection." Presented in Panel Session Entitled "Perspectives on Hypoxia." Scientific Program. May 1990. Abstract No. 353: A65

54. Botros, SM; Bruce, EN. A new mathematical model for the generation of the respiratory rhythm. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 2058-2059; 1987.

55. Bradley, GW; von Euler, C; Marttila, I; Roos, B. A Model of the Central and Reflex Inhibition of Inspiration in the Cat. Biol Cybernetics. 19: 105-116; 1975.

56. Briane, M; Quandieu, P; Henry, J; Liebaert, P. Modelisation De La Fonction Diastolique Sous Acceleration A Fort Jolt, I) Modelisation Mathematique. Medecine Aeronautique et Spatiale. 29(115): 193; 1990.

57. Brinchmann-Hansen, O; Myhre, K. Blood Pressure, Intraocular Pressure, and Retinal Vessels after High Altitude Mountain Exposure. Aviat Space Environ Med. 60: 970-976; 1989.

58. Brinchmann-Hansen, O; Myhre, K. Vascular Response of Retinal Arteries and Veins to Acute Hypoxia of 8,000, 10,000, 12,500, and 15,000 Feet of Simulated Altitude. Aviat Space Environ Med. 61: 112-116; 1990.

59. Burkhardt, WO; Adler, H; Thometz, AF; Atkinson, AJ; Ivy, AC. A Roentgenographic Study of "Bends" and "Chokes" at Altitude. Aviation Med. 16: 462-477; 1946.

60. Burns, JW; Balldin, UI. +Gz Protection with Assisted Positive-Pressure Breathing (PPB). Aerospace Research Branch, School of Aerospace Medicine, Brooks AFB, Texas; 1983.

61. Burns, JW; Balldin, UI. Assisted Positive Pressure Breathing for Augmentation of Acceleration Tolerance Time. Aviat Space Environ Med. 59: 225-233; 1988.

62. Burns, JW. Prevention of Loss of Consciousness with Positive Pressure Breathing and Supinating Seat. Aviat Space Environ Med. 59(1): 20-22; 1988.

63. Burton, AC. Physiology and Biophysics of the Circulation. Chicago, IL: Year Book Medical Publishers, Inc.; 1965.

64. Burton, DR. The Stability of Breathing Regulators. Royal Aircraft Establishment Technical Report 71229. 1971 December.

65. Burton, RR. Anti-G Suit Inflation Rate Requirements. Aviat Space Environ Med. 59(7): 601-605; 1988.

66. Burton, RR. A Conceptual Model for Predicting Pilot Group G Tolerance for Tactical Fighter Aircraft. Aviat Space Environ Med. 57: 733-744; 1986.

67. Burton, RR; Adams, JD; Dixon, GA. Differential Risks of Decompression Sickness With Varied Altitude Chamber Training Regimens. Crew Technology Division, USAF School of Aerospace Medicine, Brooks AFB, TX 78235.

68. Burton, RR; Cohen, MM; Guedry JR, FE. G-Induced Loss of Consciousness. Aviat Space Environ Med. 59(1): 1; 1988.

69. Burton, RR. G-Induced Loss of Consciousness: Definition, History, Current Status. Aviat Space Environ Med. 59(1): 2-5; 1988.

70. Burton, RR; Cohen, MM; Guedry JR, FE. G-LOC Panel: Questions, Answers, and Discussion. Aviat Space Environ Med. 1: 36-39; 1959.

71. Burton, RR. Human responses to repeated high G simulated aerial combat maneuvers. Aviat Space Environ Med. 51(11): 1185-1192; 1980.

72. Burton, RR; Shaffstall, RM. Human tolerance to aerial combat maneuvers. Aviat Space Environ Med. 51(7): 641-648; 1980.

73. Burton, RR; Leverett, SD; Michaelson, ED. Man at High Sustained +Gz Acceleration: A Review. Aerospace Med. 45(10): 1115-1136; 1974.

74. Burton, RR. Physiologic Bases for Increased G-Level and G-Duration Tolerances. USAF School of Aerospace Medicine, Brooks Air Force Base, TX 78235. 1990 February.

75. Byrne, JC; McIlwain-Axten, CJ. An Analytical Model of the Aircrew Oxygen Breathing System. SAFE Journal. 22(6): 8-14; 1992.

76. Capek, JM; McCormick, M; Roy, RJ. Noninvasive Estimation of Cardiac Output Using a Differential CO2 Flick Equation. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 1030-1031; 1987.

77. Carley, D; Maayan, C; Grimes, J; Shannon, DC. Oscillations in Respiratory Parameters During Periodic Breathing. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 2062-2063; 1987.

78. Chang, HK; Tai, RC; Farhi, LE. Some Implications of Ternary Diffusion in the Lung. Respir Physiol. 23: 109-120; 1975.

79. Chatterjea, A; Libll, JN. A Model for the Prediction of Pressure Pulse Contours in the Ascending Aorta. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 868-869; 1987.

80. Chatwin, PC. On the longitudinal dispersion of passive contaminant in oscillatory flows in tubes. J Fluid Mech. 71(3): 513-527; 1975.

81. Christensen, P; Nielsen, OW; Gronlund, J. Nonlinear Curve Fitting Improves Noise Sensitivity of the Single-Breath Method for Estimation of Cardiac Output. Aviat Space Environ Med. 60: 803-806; 1989.

82. Collins, E; Calvert, RE; Hardy, HH. Mathematical Simulation of the Cardiopulmonary System. AFOSR-76-2905; AD-A215 474. 99 pp; 1979 December.

83. Collins, RE; Calvert, RE; Hardy, HH. Mathematical Simulation of the Cardiopulmonary System. AFOSR-TR-89-1571. 99 pp; 1979 December.

84. Comens, P; Reed, D; Mette, M. Physiologic Responses of Pilots Flying High-Performance Aircraft. Aviat Space Environ Med. 58: 205-210; 1987.

85. Craig, FN; Blevins, WV; Cummings, EG. Exhausting work limited by external resistance and inhalation of carbon dioxide. J Appl Physiol. 29(6): 847-851: 1970.

86. Croston, RC; Rummel, JA; Kay, FJ. Computer Model of Cardiovascular Control System Responses to Exercise. J Dynam Sys Meas Cntrl. 301-307; 1973 September.

87. Cumming, G; Crank, J; Horsfield, K; Parker, I. Gaseous Diffusion in the Airways of the Human Lung. Respir Physiol. 1: 58-74; 1966.

88. Cumming, G; Horsfield, K; Jones, JG; Muir, DCF. The Influence of Gaseous Diffusion on the Alveolar Plateau at Different Lung Volumes. Respir Physiol. 2: 386-398; 1967.

89. Dahlback, GO; Jonson, B; Lundgren, EG. Influence of Hydrostatic Thorax Compression and Intra-Thoracic Blood Pooling on Dynamic Lung Mechanics During Head-Out Immersion. Study was supported by the National Defense Research Institute (Project No. FOA 098-5123:7), the Swedish National Association against Chest and Heart Diseases, and the Swedish Medical Research Council (Grant 14X-02872). 1-17; 1978.

90. Dahlback, GO. Lung Mechanics During Immersion In Water with special reference to pulmonary air trapping. From the Laboratory of Aviation and Naval Physiology, Institute of Physiology and Biophysics, University of Lund, Sweden. 1-62; 1978.

91. Dahlback, GO; Balldin, UI. Pulmonary atelectasis formation during diving with closed-circuit oxygen breathing apparatus. Undersea Biomed Res. 12(2): 129-137; 1985.

92. Dahlback, GO. Respiratory Load Imposed by the Diver's Protective Equipment. IN: Environmental Ergonomics - Sustaining Human Performance in Harsh Environments. Eds. Mekjavic, IB, Banister, EW, Morrison, JB; New York: Taylor & Francis; 1988; Chapter 18: 332-339.

93. Daubenspec, JA; Harver, A. Identification of Respiratory Neural Reflexes Using a Neuromechanical Model. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society; 2054-2055; 1987.

94. Davidson, MF; Fitz-Gerald, JM. Flow patterns in models of small airway units of the lung. J Fluid Mech. 52(1): 161-177; 1972.

95. Davidson, MR. The Influence of Gas Exchange on Lung Gas Concentrations During Air Breathing. Bull Math Biol. 39: 73-86; 1977.

96. Davidson, MR. Lung Gas Mixing During Expiration Following an Inspiration of Air. Bull Math Biol. 37: 113-126; 1975.

97. Davidson, MR; Fitz-Gerald, JM. Transport of O_2 Along a Model Pathway Through the Respiratory Region of the Lung. Bull Math Biol. 36: 275-303; 1974.

98. De Meersman, RE. A Program in Quickbasic for the Estimation of Cardiac Output. Comput Biol Med. 18(4): 241-243; 1988.

99. De Vries, RP; Luijendijk, SCM; Zwart, A. Helium and sulfur hexafluoride washout in asymmetric lung models. J Appl Physiol. 51(5): 1122-1130; 1981.

100. Dennis, DR; Gale, GG. An Investigation of the Factors Affecting the Stability of a Miniature Oxygen/Airmix Breathing Regulator. Royal Aircraft Establishment Technical Report 73136. 1-54; 1973 November.

101. Dennis, DR. Notes on the Design and Development of a Breathing Simulator of Improved Performance for Use in the Dynamic Testing of Aircrew Breathing Systems (Beaver Replacement). Royal Aircraft Establishment Tech Memo EP 507. 1-25; 1972 June.

102. Design Report for the Multi-Man Oxygen Concentrator. Prepared for Naval Air Development Center, Warminster, PA 18974 under Contract #N62269-83-R-0380; Pub. No. 8170; by Clifton Precision Instruments & Life Support Division; 1984 August 7.

103. Dixon, GA; Krutz Jr, RW; Fischer, JR. Decompression Sickness and Bubble Formation in Females Exposed to a Simulated 7.8 psia Suit Environment. Aviat Space Environ Med. 59: 1146-1149; 1988.

104. DuBois, AB; Brody, AW; Lewis, DH; Burgess Jr, BF. Oscillation Mechanics of Lungs and Chest in Man. J Appl Physiol. 8: 587-594; 1956 May.

105. Dubois, AB. Resistance to breathing. IN: Handbook of Physiology, Section 3: Respiration. Eds. Fenn, WO; Rahn, H. Baltimore, MD: Williams & Wilkins; 2(Chap. 16): 451-463; 1964-65.

106. Elad, D; Kamm, RD; Shapiro, AH. Mathematical Simulation of Forced Expiration. J Appl Physiol. 65(1): 14-25; 1988 July.

107. Engel, LA; Paiva, M. Analyses of Sequential Filling and Emptying of the Lung. Respir Physiol. 45: 309-321; 1981.

108. Engel, LA; Grassino, A; Anthonisen, NR. Demonstration of airway closure in man. J Appl Physiol. 38(6): 1117-1125; 1975.

109. Engel, LA; Paiva, M; Siegler, DIM; Fukuchi, Y. Dual Tracer Single Breath Studies of Gas Transport in the Lung. Respir Physiol. 36: 103-119; 1979.

110. Engel, LA; Menkes, H; Wood, LDH; Utz, G; Joubert, J; Macklem, PT. Gas mixing during breath holding studied by intrapulmonary gas sampling. J Appl Physiol. 35(1): 9-17; 1973.

111. Engel, LA. Gas mixing within the acinus of the lung. J Appl Physiol. 54(3): 609-618; 1983.

112. Engel, LA; Utz, G; Wood, LDH; Macklem, PT. Ventilation distribution in anatomical lung units. J Appl Physiol. 37(2): 194-200; 1974.

113. Ernsting, J. The Effects of Resistance to Respiration in Man - A Review of the Literature. Royal Air Force Institute of Aviation Medicine; IAM Report #112; 1959 January.

114. Ernsting, J. The Genesis and Reduction of Resistance to Breathing in Aircraft Oxygen Delivery Systems. Royal Air Force Institute of Aviation Medicine, Farnborough, Hants, England GU14 6SZ. 1-4.

115. Ernsting, J. High altitude partial pressure suits. Technological News - The Journal of Normalair - Garrett Limited. Summer: 14-19; 1979.

116. Ernsting, J. The Ideal Relationship Between Inspired Oxygen Concentration and Cabin Altitude. Aerospace Medicine. 34(11): 991-997; 1963.

117. Ernsting, J. Inspiratory Flow Patterns in Subjects at Rest and During Speech. Flying Personnel Research Committee, Air Ministry, Royal Air Force Institute of Aviation Medicine, Farnborough; MODUK (Air) Flying Personnel Research Committee Report #1144. 1-11; 1960 October.

118. Ernsting, J. The Physiology of Pressure Breathing. IN: A textbook of aviation physiology. Ed. Gillies, JA. Oxford:Pergamon Press. 1st ed.; Chapter 17: 343-373; 1965.

119. Ernsting, J. Prevention of Hypoxia-Acceptable Compromises. Aviat Space Environ Med. 49(5): 495-502; 1978.

120. Ernsting, J; Roxburgh, HL; Wagner, PR. Rapid Decompression in the Helmet, Jerkin, Anti-G Suit System - A Preliminary Report. Flying Personnel Research Committee, Air Ministry, Royal Air Force Institute of Aviation Medicine, Farnborough; MODUK (Air) Flying Personnel Research Committee Report #1150. 1-44; 1960 April.

121. Estock, BC; Gray, GW; Ford, GS. Evaluation with Doppler Ultrasonic Bubble Detection of Standard Canadian Forces Aeromedical Training Altitude Profiles. Department of National Defence - Canada, Defence and Civil Institute of Environmental Medicine, DCIEM Technical Communication #83-C-07. 1-17; 1983 February.

122. Ewan, PW; Jones, HA; Nosil, J; Obdrzalek, J; Hughes, JMB. Uneven Perfusion and Ventilation Within Lung Regions Studied with Nitrogen-13. Respir Physiol. 34: 45-59; 1978.

123. Eyles, JG; Pimmel, RI. Estimating Respiratory Mechanical Parameters in Parallel Compartment Models. IEEE Trans Biomed Eng. BME-28(4): 313-317; 1981.

124. Eyles, JG; Pimmel, RL; Fullton, JM; Bromberg, PA. Parameter Estimates in a Five-Element Respiratory Mechanical Model. IEEE Trans Biomed Eng. BME-29(6): 460-463; 1982.

125. Filuk, RB; Anthonisen, NR. Changes in regional emptying sequence need not change maximum expiratory flow. J Appl Physiol. 60(6): 1834-1838; 1986.

126. Fleischer, LS; Bridge, JF. An Analytical Model of Inspiratory Gas Flow Within the Human Bronchial Tree. American Society of Mechanical Engineers Winter Annual Meeting; November 17-22, 1974; New York, NY. Paper #74-WA/Bio-13. 1-15; 1974.

127. Frazer, DG; Franz, GN. Trapped Gas and Lung Hysteresis. Respir Physiol. 46: 237-246; 1981.

128. Fredberg, JJ. Augmented diffusion in the airways can support pulmonary gas exchange. J Appl Physiol. 49(2): 232-238; 1980.

129. Fredberg, JJ; Mead, J. Impedance of intrathoracic airway models during low-frequency periodic flow. J Appl Physiol. 47(2): 347-351; 1979.

130. Fry, DL; Hyatt, RE; McCall, CB; Mallos, AJ. Evaluation of Three Types of Respiratory Flowmeters. J Appl Physiol. 10(2): 210-214; 1957.

131. Fryer, DI. Subatmospheric Decompression Sickness in Man. Slough, Eng: Technivision services. 40-41; 108-109; 112-117; 186-187; 259; 262-263; 270-275; 1969.

132. Fukaya, H; Martin, CJ; Young, AC; Katsura, S. Mechanical properties of alveolar walls. J Appl Physiol. 25(6): 689-695; 1968.

133. Fukuchi, Y; Roussos, CS; Macklem, PT; Engel, LA. Convection, Diffusion and Cardiogenic Mixing of Inspired Gas in the Lung; An Experimental Approach. Respir Physiol. 26(77-90); 1976.

134. Fukuchi, Y; Cosio, M; Kelly, S; Engel, LA. Influence of pericardial fluid on cardiogenic gas mixing in the lung. J Appl Physiol. 42(1): 5-12; 1977.

135. Fukuchi, Y; Cosio, M; Murphy, B; Engel, LA. Intraregional Basis for Sequential Filling and Emptying of the Lung. Respir Physiol. 41: 253-266; 1980.

136. Fung, YC. A Theory of Elasticity of the Lung. Transactions of the American Society of Mechanical Engineers; Paper No. 73-APM-R. J Appl Mech. 1-7; 1973.

137. Fung, YC. Biomechanics. New York: Springer-Verlag; 1993.

138. Fung, YC; Sobin, SS. Elasticity of the pulmonary alveolar sheet. Circ Res. 30: 451-469; 1972.

139. Gaffie, D; Quandieu, P; Liebaert, PH; Cohen-Zardy, D; Daumas, T; Guillaume, A. Circulatory Biomechanics Effects of Acceleration. IN: High Altitude and High Accelerations: Protection for Military Aircrew. Neuilly-sur-Seine, France: Advsiory Group for Aerospace Research and Development (AGARD). 15 pp; 1991 October.

140. Gale, GE; Torre-Bueno, JR; Moon, RE; Saltzman, HA; Wagner, PD. Ventilation-perfusion inequality in normal humans during exercise at sea level and simulated altitude. J Appl Physiol. 58(3): 978-988; 1985.

141. Gale, GG. Preliminary Acceptance Tests of a Breathing Simulator for Research, Development and Dynamic Testing of Aircrew Breathing Equipment. Royal Aircraft Establishment Tech Memo EP 458. 1-31; 1970 October.

142. Gallagher, RR. Evaluation of Simulation Capabilities with a Respiratory-Circulatory System Integration Scheme. IN: Modeling and simulation. (A75-24745 10-66) Pittsburgh, PA: Instrument Society of America; 1974. 5(Part 1): 483-487.

143. Gillies, JA. A textbook of aviation physiology. Oxford: Pergamon Press; 1965; 1st ed.: 130-131.

144. Gillingham, KK. High-G Stress and Orientational Stress: Physiologic Effects of Aerial Maneuvering. Aviat Space Environ Med. 59(11, Suppl): A10-20; 1988.

145. Gillingham, KK; Fosdick, JP. High-G Training for Fighter Aircrew. Aviat Space Environ Med. 59(1): 12-19; 1988.

146. Gillingham, KK; McNee, RC. Mathematical Modeling of Arterial Oxygen Saturation and Eye-Level Blood Pressure During +Gz Stress. Proceedings of the AGARD Conference #253; NATO Advisory Group for Aerospace Research & Development. A15-1 - A15-7; 1979 June.

147. Gillingham, KK; Freeman, JJ; McNee, RC. Transfer Functions for Eye-Level Blood Pressure During +Gz Stress. Aviat Space Environ Med. 48(11): 1026-1034; 1977.

148. Glaister, DH; Schroter, RC; Sudlow, MF; Milic-Emili, J. Bulk Elastic Properties of Excised Lungs and the Effect of a Transpulmonary Pressure Gradient. Respir Physiol. 17: 347-364; 1973.

149. Glaister, DH. Current and Emerging Technology in G-LOC Detection: Noninvasive Monitoring of Cerebral Microcirculation Using Near Infrared. Aviat Space Environ Med. 59(1): 23-28; 1988.

150. Glaister, DH; Schroter, RC; Sudlow, MF; Milic-Emili, J. Transpulmonary Pressure Gradient and Ventilation Distribution in Excised Lungs. Respir Physiol. 17: 365-385; 1973.

151. Glazier, JB; Hughes, MB; Maloney, JE; West, JB. Measurements of capillary dimensions and blood volume in rapidly frozen lungs. J Appl Physiol. 26(1): 65-76; 1969

152. Glenny, RW; Polissar, L; Robertson, HT. Relative Contribution of Gravity to Pulmonary Perfusion Heterogeneity. J Appl Physiol. 71(6): 2449-2452; 1991 December.

153. Granger, WM; Miller, DA; Ehrhart, IC; Hofman, WF. The Effect of Blood Flow and Diffusion Impairment on Pulmonary Gas Exchange: A Computer Model. Comput Biomed Res. 20: 497-506; 1987.

154. Grant, BJB; Schneider, AM. Dynamic Response of Local Pulmonary Blood Flow to Alveolar Gas Tensions: Analysis. J Appl Physiol. 54(2): 445-452; 1983 February.

155. Grassino, AE; Anthonisen, NR. Chest wall distortion and regional lung volume distribution in erect humans. J Appl Physiol. 39(6): 1004-1007; 1975.

156. Grassino, AE; Forkert, L; Anthonisen, NR. Configuration of the Chest Wall During Increased Gravitational Stress in Erect Humans. Respir Physiol. 33: 271-278; 1978.

157. Gray, JS. Effect of Denitrogenation at Various Altitudes on Aeroembolism in Cadets. Research Report. AAF School of Aviation Medicine, Randolph Field, Texas; Project #216, Report #1. 1-8; 1944 January.

158. Gray, JS. Present Status of the Problem of Decompression Sickness. Review I. The Research Laboratory. The Army Air Forces School of Aviation Medicine, Randolph Field, Texas; Project #458, Report #1. 29-47; 1944 October.

159. Gray, JS. The Time-Distribution of Symptoms at 35,000 and 38,000 feet in the Low Pressure Chamber. Research Report. The School of Aviation Medicine, Randolph Field, Texas; Project #71, Report #1. 1-6; 1942 September.

160. Green, M; Mead, J; Turner, JM. Variability of maximum expiratory flow-volume curves. J Appl Physiol. 37(1): 67-74; 1974.

161. Guyton, AC. Textbook of Medical Physiology. 7th Ed. Philadelphia, PA: W.B. Saunders; 1985.

162. Halperin, HR; Tsitlik, JE. Programmable Pneumatic Generator for Manipulation of Intrathoracic Pressure. IEEE Trans Biomed Eng. BME-34(9): 738-742; 1987.

163. Hansen, JE; Ampaya, EP; Bryant, GH; Navin, JJ. Branching pattern of airways and air spaces of a single human terminal bronchiole. J Appl Physiol. 38(6): 983-989; 1975.

164. Hansen, JE; Ampaya, EP. Human air space shapes, sizes, areas, and volumes. J Appl Physiol. 38(6): 990-995; 1975.

165. Hantos, Z; Daroczy, B; Suki, B; Galgoczy, G; Csendes, T. Forced oscillatory impedance of the respiratory system at low frequencies. J Appl Physiol. 60(1): 123-132; 1986.

166. Harding, RM. Respiratory Physiology During Light - A Review. Royal Air Force Institute of Aviation Medicine, Farnborough, Hampshire. IAM Report #642; 1985 June.

167. Heijmans, F; Bert, JL; Pinder, KL. Digital Simulation of Pulmonary Microvascular Exchange. Comput Biol Med. 16(2): 69-90; 1986 February.

168. Heijmans, F; Bert, J; Pinder, K. Alveolar Flooding: A Computer Simulation. Comput Methods Programs Biomed. 23(2): 93-101; 1986 October.

169. Hesser, CM; Lind, F. Role of airway resistance in the control of ventilation during exercise. Acta Physiol Scand. 120: 557-565; 1984.

170. Hesser, CM; Lind, F. Ventilatory and Occlusion-Pressure Responses to Incremental-Load Exercise. Respir Physiol. 51: 391-401; 1983.

171. Hesser, CM; Linnarsson, D; Fagraeus, L. Pulmonary mechanics and work of breathing at maximal ventilation and raised air pressure. J Appl Physiol. 50(4): 747-753; 1981.

172. Hey, EN; Lloyd, BB; Cunningham, DJC; Jukes, MGM; Bolton, DPG. Effects of Various Respiratory Stimuli on the Depth and Frequency of Breathing in Man. Respir Physiol. 1: 193-205; 1966.

173. Holden, RD; Bomar, JB; O'Connor, RB; Wright, CS; Nesthus, TE. Acceptability of Standard USAF Breathing Gear at High Altitude. Proceedings of the 25th Annual SAFE Symposium. 166-170; 1987 November.

174. Holden, RD; Ernsting, J; Baumgardner, FW. Physiological Assessment of Current USAF Integrated Oxygen Delivery Components. Crew Protection Branch, Crew Technology Division, USAF School of Aerospace Medicine, Brooks AFB, TX 78235. 127-128; 1980. (Note: Preprint of 1980 Annual Scientific Meeting of Aerospace Medical Association.)

175. Holness, DE; Porlier, JAG; Ackles, KN; Wright, GR. Respiratory Gas Exchange During Positive Pressure Breathing and Rapid Decompression to Simulated Altitudes of 18.3 and 24.4 km. Aviat Space Environ Med. 51(5): 454-458; 1980.

176. Holtz, B; Bake, B; Oxhoj, H. Effect of inspired volume on closing volume. J Appl Physiol. 41(5): 623-630; 1976.

177. Homer, LD; Weathersby, PK. How well mixed is inert gas in tissues? J Appl Physiol. 60(6): 2079-2088; 1986.

178. Horvath, SM. Recent Contributions to High Altitude Physiology. IN: Man in Stressful Environments: thermal and work physiology. Eds. Shiraki, K; Yousef, M. Springfield, IL: C.C.Thomas. Chap. 10: 159-166; 1987.

179. Hrebien, L. Current and Emerging Technology in G-LOC Detection: Pulse Wave Delay for +Gz Tolerance Assessment. Aviat Space Environ Med. 59(1): 29-31; 1988.

180. Hughes, JMB; Hoppin Jr, FG; Mead, J. Effect of lung inflation on bronchial length and diameter in excised lungs. J Appl Physiol. 32(1): 25-35; 1972.

181. Hughes, JMB; Jones, HA; Wilson, AG; Grant, BJB; Pride, NB. Stability of intrapulmonary bronchial dimensions during expiratory flow in excised lungs. J Appl Physiol. 37(5): 684-694; 1974.

182. Hughes, JMB; Rosenzweig, DY; Kivitz, PB. Site of airway closure in excised dog lungs: Histologic demonstration. J Appl Physiol. 29: 340-344; 1970.

183. Hunt, B. Diffusion in Laminar Pipe Flow. Int J Heat Mass Transfer. 20: 393-401; 1977.

184. Hyatt, RE; Zimmerman, IR; Peters, GM; Sullivan, WJ. Direct writeout of total respiratory resistance. J Appl Physiol. 28(5): 675-678; 1970.

185. Hyatt, RE. Expiratory flow limitation. J Appl Physiol. 55(1): 1-8; 1983.

186. Hyatt, RE; Flath, RE. Influence of lung parenchyma on pressure-diameter behavior of dog bronchi. J Appl Physiol. 21(5): 1448-1452; 1966.

187. Hyatt, RE; Okeson, GC; Rodarte, JR. Influence of expiratory flow limitation on the pattern of lung emptying in normal man. J Appl Physiol. 35(3): 411-419; 1973.

188. Isabey, D; Chang, HK. Steady and unsteady pressure-flow relationships in central airways. J Appl Physiol. 51(5): 1338-1348; 1981.

189. Jackson, AC; Lutchen, KR. Modeling of respiratory system impedances in dogs. J Appl Physiol. 62(2): 414-420; 1987.

190. Jaron, D; Moore, TW; Bai, J. Cardiovascular Responses to Acceleration Stress: A Computer Simulation. Proceedings of the IEEE. 76(6): 700-707; 1988.

191. Jaron, D; Moore, TW; Chu, CL. A Cardiovascular Model for Studying Impairment of Cerebral Function During +Gz Stress. Aviat Space Environ Med. 55(1): 24-31; 1984.

192. Jaron, D; Moore, TW; Vieyres, P. A Cardiovascular Model of G-Stress Effects: Preliminary Studies with Positive Pressure Breathing. IN: High Altitude and High Acceleration: Protection for Military Aircrew. Neuilly-sur-Seine, France: Advisory Group for Aerospace Research and Development (AGARD). 7 pp.; 1991 October

193. Johanson, DC; Pheeny, HT. A New Look at the Loss of Consciousness Experience Within the U.S. Naval Forces. Aviat Space Environ Med. 59(1): 6-8; 1988.

194. Juznic, G; Emri, I; Struna, B; Peterec, D; Knap, B. Simulating the Strength of the Heart Beat. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 897-898; 1987.

195. Kakitsuba, N; Michina, H; Mekjavic, IB. Clothing Surface Area as Related to Body Volume and Clothing Microenvironment Volume. Aviat Space Environ Med. 58: 411-416; 1987.

196. Kallok, MJ; Wilson, TA; Rodarte, JR; Lai-Fook, SJ; Chevalier, PA; Harris, LD. Distribution of regional volumes and ventilation in excised canine lobes. J Appl Physiol. 47(1): 182-191; 1979.

197. Kapitan, KS; Hempleman, SC. Computer Simulation of Mammalian Gas-Exchange. Comput Biol Med. 16(2): 91-101; 1986 February.

198. Kappos, AD; Rodarte, JR; Lai-Fook, SJ. Frequency dependence and partitioning of respiratory impedance in dogs. J Appl Physiol. 51(3): 621-629; 1981.

199. Kar, S. Investigation of Blood Flow in Veins. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 864-865; 1987.

200. Karemaker, JM; Akkerman, EM; Vanleeuwen, M; Wesseling, KH; Settels, JJ; Wieling, W; Dambrink, JHA. Control of Blood Pressure in Humans under Microgravity. IN: Space Research Organization Netherlands, Activities Report of the Space Research Organization Netherlands. Utrecht, Netherlands: Space Research Organization. 125-128; 1991 April.

201. Khoo, MC; Kronauer, RE; Strohl, KP; Slutsky, AS. Factors Inducing Periodic Breathing in Humans: A General Model. J Appl Physiol. 53(3): 644-659; 1982 September.

202. Khoo, MCK. ARMA Modeling of Gas Exchange During Spontaneous Breathing. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 2060-2061; 1987.

203. Khoo, MCK; Kronauer, RE; Strohl, KP; Slutsky, AS. Factors inducing periodic breathing in humans: a general model. J Appl Physiol. 53(3): 644-659; 1982.

204. Kilburn, KH; Sieker, HO. Hemodynamic Effects of Continuous Positive and Negative Pressure Breathing in Normal Man. Circulation Research. 8: 660-669; 1960 May.

205. Kimball, WR; Kelly, KB; Mead, J. Thoracoabdominal blood volume change and its effect on lung and chest wall volumes. J Appl Physiol. 61(3): 953-959; 1986.

206. King, M; Chang, HK; Weber, ME. Resistance of mucus-lined tubes to steady and oscillatory airflow. J Appl Physiol. 52(5): 1172-1176; 1982.

207. Kinker, JR; Haffor, AS; Stephan, M; Clanton, TL. Kinetics of CO Uptake and Diffusing Capacity in Transition from Rest to Steady-State Exercise. J Appl Physiol. 72(5): 1764-1772; 1992 May.

208. Kjellmer, I; Sandqvist, L; Berglund, E. 'Alveolar plateau' of the single breath nitrogen elimination curve in normal subjects. J Appl Physiol. 14(1): 105-108; 1959.

209. Knudson, RJ; Kaltenborn, WT. Evaluation of Lung Elastic Recoil by Exponential Curve Analysis. Respir Physiol. 46: 29-42; 1981.

210. Kobayashi, H; Abe, T; Kawashiro, T; Tanabe, K; Yokoyama, T. Estimation of the Distribution Profile of Airway Resistance in th Lungs. Comput Biomed Res. 20: 507-525; 1987.

211. Krueger, JJ; Bain, T; Patterson Jr, JL. Elevation gradient of intrathoracic pressure. J Appl Physiol. 16(3): 465-468; 1961.

212. Krutz Jr, RW; Sears, WJ; Gould Jr, KG; Bancroft, RW. Aeromedical evaluation of the phased-dilution concept for oxygen breathing systems. IN: Current Status in Aerospace Medicine. NATO Advisory Group for Aerospace Research and Development (AGARD); AGARD Conference Pre-Print No. 110 - AGARD-CPP-110. C8-1 - C8-7; 1974.

213. Krutz Jr, RW; Dixon, GA; Harvey, WT. Minimum Pressure for a Zero-Prebreathe Pressure Suit. Proceedings of the Fifteenth Intersociety Conference on Environmental Systems; July 15-17, 1985; San Francisco, CA. SAE Technical Paper Series - 851315. Warrendale, PA: Society of Automotive Engineers; 1985: 1-3.

214. Lacquet, LM; Van Der Linden, LP; Paiva, M. Transport of H2 and SF6 in the Lung. Respir Physiol. 25: 157-173; 1975.

215. La Force, RC; Lewis, BM. Diffusional transport in the human lung. J Appl Physiol. 28(3): 291-298; 1970.

216. Lai-Fook, SJ; Kallok, MJ. Bronchial-arterial interdependence in isolated dog lung. J Appl Physiol. 52(4): 1000-1007; 1982.

217. Lambert, RK. Analysis of bronchial mechanics and density dependence of maximal expiratory flow. J Appl Physiol. 61(1): 138-149; 1986.

218. Lambert, RK; Wilson, TA. A model for the elastic properties of the lung and their effect on expiratory flow. J Appl Physiol. 34(1): 34-48; 1973.

219. Lambert, RK; Wilson, TA; Hyatt, RE; Rodarte, JR. A Computational Model for Expiratory Flow. J Appl Physiol. 52(1): 44-56; 1982 January.

220. Landervik, B; Dahlback, G. Man rating of the Anti-G protection in a human centrifuge of the Tactical Flight Combat Suit (TRCS), the EROS oxygen regulator/anti-G valve with Swedish JAS 39 test pilots (11 Appendix). Forsvarets Materielverk Test Report. 1-35; 1991.

221. Langhorst, P; Lambertz, M; Schulz, G. Dynamic Organization of the Central Nervous System for Cardiovascular, Respiratory and Somatomotor Regulation. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 891-892; 1987.

222. Laviolette, M; Cormier, Y. Intra Versus Interregional Nitrogen Gradients in the Single Breath Nitrogen Test. Respir Physiol. 41: 267-277; 1980.

223. Leblanc, P; Ruff, F; Milic-Emili, J. Effects of age and body position on "airway closure" in man. J Appl Physiol. 28(4): 448-451; 1970.

224. Lemen, RJ; Gerdes, CB; Wegmann, MJ; Perrin, KJ. Frequency spectra of flow and volume events for forced vital capacity. J Appl Physiol. 53(4): 977-984; 1982.

225. Lenox, JB; Koegel, E. Evaluation of a new low-resistance breathing valve. J Appl Physiol. 37(3): 410-413; 1974.

226. Lichtenstein, O; Ben-Haim, SA; Saidel, GM; Dinnar, U. Role of the Diaphragm in Chest Wall Mechanics. J Appl Physiol. 72(2): 568-574; 1992 February.

227. Lim, LL. A Statistical Model of the VA/Q Distribution. J Appl Physiol. 69(1): 281-292; 1990 July.

228. Lind, F; Hesser, CM. Breathing pattern and lung volumes during exercise. Acta Physiol Scand. 120: 123-129; 1984.

229. Lind, F; Hesser. Breathing pattern and occlusion pressure during moderate and heavy exercise. Acta Physiol Scand. 122: 61-69; 1984.

230. Lind, FG; Truve, AB; Lindborg, PO. Microcomputer-assisted on-line measurement of breathing pattern and occlusion pressure. J Appl Physiol. 56(1): 235-239; 1984.

231. Lind, FG. Respiratory Drive and Breathing Pattern during Exercise in Man. Acta Physiol Scand. Supplementum 533: 1-47; 1984.

232. Linehan, JH; Dawson, CA; Rickaby, DA; Bronikowski, TA. Pulmonary vascular compliance and viscoelasticity. J Appl Physiol. 61(5): 1802-1814; 1986.

233. Linehan, JH; Haworth, ST; Nelin, LD; Krenz, GS; Dawson, CA. A Simple Distensible Vessel Model for Interpreting Pulmonary Vascular Pressure-Flow Curves. J Appl Physiol. 73(3): 987-994; 1992 September.

234. Liu, S; Wilson, TA; Schreiner, K. Gravitational Forces on the Chest Wall. J Appl Physiol. 70(4): 1506-1510; 1991 April.

235. Loeppky, JA; Caprihan, A; Luft, UC. Va/Q Inequality During Clinical Hypoxemia and its Alterations. IN: Man in Stressful Environments: thermal and work physiology. Eds. Shiraki, K, Yousef, M. Springfield, IL: C.C. Thomas. Proceedings of the International Symposium on Physiology of Stressful Environments; September 21-24, 1986; Japan; sponsored by the IUPS Commission on Environmental Physiology. Chapter 13: 199-232; 1987.

236. Loeppky, JA; Fletcher, ER; Myhre, LG; Luft, UC. Validation and Application of Single Breath Cardiac Output Determinations in Man. Aviat Space Environ Med. 57: 759-768; 1986.

237. Lorino, H; Lorino, AM; Harf, A; Atlan, G; Laurent, D. Linear Modeling of Ventilatory Mechanics during Spontaneous Breathing. Comput Biomed Res. 15: 129-144; 1982.

238. Lorino, J; Mariette, C; Lorino, AM; Harf, A. Four and Six-Parameter Models of Forced Random Noise Respiratory Impedance. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 1807-1808; 1987.

239. Luft, UC; Clamann, HG; Adler, HF. Alveolar Gases in Rapid Decompression to High Altitudes. J Appl Physiol. 2: 37-48; 1949 July.

240. Luijendijk, SCM; Zwart, A; de Vries, WR; Salet, WM. The Sloping Alveolar Plateau at Synchronous Ventilation. Pflugers Archiv European Journal of Physiology. 384: 267-277; 1980.

241. Lutchen, KR; Merer, DM. Optimal Experiment Design for estimating Mechanical Properties from Respiratory Impedance Data. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 1809-1810; 1987.

242. Lutchen, KR; Jackson, AC. Reliability of parameter estimates from models applied to respiratory impedance data. J Appl Physiol. 62(2): 403-413; 1987.

243. Lutchen, KR; Jackson, AC. Statistical Measures of Parameter Estimates from Models Fit to Respiratory Impedance Data: Emphasis on Joint Variabilities. IEEE Trans Biomed Eng. BME-33(11): 1000-1009; 1986.

244. Marbarger, JP; Kadetz, W; Paltarokas. J; Yariakojis, D; Hansen, J; Dickinson, J. Gaseous Nitrogen Elimination at Ground Level and Simulated Altitude and the Occurrence of Decompression Sickness. Air University, School of Aviation Medicine, USAF; Randolph AFB, Texas. 1-21; 1956 February.

245. Marotte, H; Toure, C; Clere, JM; Vieillefond, H. Rapid Decompression of a Transport Aircraft Cabin: Protection Against Hypoxia. Aviat Space Environ Med. 61: 21-27; 1990.

246. Martin, CJ; Das, S; Young, AC. Terminal nitrogen rise. J Appl Physiol. 41(4): 517-522; 1976.

247. Matalon, S; Dashkoff, N; Nesarajah, MS; Klocke, FJ; Farhi, LE. Effects of hyperventilation on pulmonary blood flow and recirculation time of humans. J Appl Physiol. 52(5): 1161-1166; 1982.

248. Matalon, S; Nesarajah, MS; Farhi, LE. Pulmonary and circulatory changes in conscious sheep exposed to 100% O2 at 1 ATA. J Appl Physiol. 53(1): 110-116; 1982.

249. McCall, CB; Hyatt, RE; Noble, FW; Fry, DL. Harmonic Content of Certain Respiratory Flow Phenomena of Normal Individuals. J Appl Physiol. 10(2): 215-218; 1957.

250. McIlroy, MB; Hargrave, VK; Targett, RC. A Model of the Pulmonary Arterial Bed in Adults and Infants. Comput Biomed Res. 23(2): 130-138; 1990 April.

251. Mead, J; Agostoni, E. Dynamics of breathing. IN: Handbook of Physiology, Section 3: Respiration. Eds. Fenn, WO; Rahn, H. Baltimore, MD: Williams & Wilkins; 2(Chap. 14): 411-427; 1964-65.

252. Mead, J; Takishima, T; Leith, D. Stress distribution in lungs: A model of pulmonary elasticity. J Appl Physiol. 28(5): 596-608; 1970.

253. Mead, J; Whittenberger, JL; Radford Jr, EP. Surface Tension as a Factor in Pulmonary Volume-Pressure Hysteresis. J Appl Physiol. 10(2): 191-196; 1957.

254. Mead, J; Milic-Emili, J. Theory and methodology in respiratory mechanics with glossary of symbols. IN: Handbook of Physiology, Section 3: Respiration. Eds. Fenn, W;, Rahn, H. Baltimore, MD: Williams & Wilkins; 2(Chap. 11): 363-376; 1964-65.

255. Meader, WL. Decompression Sickness in High-altitude Flight. Aerospace Medicine. 301-303; 1967 March.

256. Meyer, M; Hook, C; Rieke, H; Piiper, J. Gas mixing in dog lungs studied by single-breath washout of He and SF6. J Appl Physiol. 55(6): 1795-1802; 1983.

257. Michels, DB; West, JB. Distribution of pulmonary ventilation and perfusion during short periods of weightlessness. J Appl Physiol. 45(6): 987-998; 1978.

258. Mickolls, PM; Hensley, MJ. Technique for Assessing the Response of the Respiratory Controller to Hypoxia and Hypercapnia. J BIomed Eng. 8: 305-312; 1986 October.

259. Milic-Emili, J; Henderson, JAM; Dolovich, MB; Trop, D; Kaneko, K. Regional distribution of inspired gas in the lung. J Appl Physiol. 21: 749-759; 1966.

260. Milic-Emili, J; Mead, J; Turner, JM. Topography of esophageal pressure as a function of posture in man. J Appl Physiol. 19(2): 212-216; 1964.

261. Millette, B; Robertson, PC; Ross, WRD; Anthonisen, NR. Effect of expiratory flow rate on emptying of lung regions. J Appl Physiol. 27(5): 587-591; 1969.

262. Montgomery, LD; Hanish, HM; Burns, JW. A System to Measure Lower body Volume Changes During Rapid Onset High-G Acceleration. Aviat Space Environ Med. 59: 1098-1102; 1988.

263. Moore, TW; Jaron, D. Cardiovascular Model for Studying Impairment Under Acceleration. IEEE Engineering in Medicine and Biology. 37-40; 1991 March.

264. Morgan, TR; Baumgardner, FW; Crigler, JC; Reid, DH; Tays, MA. Preliminary Analysis of Available Inflight Respiratory Data. USAF School of Aerospace Medicine, Aerospace Medical Division (AFSC), Brooks AFB, TX 78235; Interim Report for Period January 1976-May 1977; Report SAM-TR-77-20; Dec 1977.

265. Morgan, TR; Reid, DH; Baumgardner, FW. Pulmonary Ventilation Requirements Evident in the Operaiton of Representative High-Performance Aircraft.

Environmental Sciences Division, USAF School of Aerospace Medicine, Brooks AFB, TX. Report SAM-TR. 156-157.

266. Mosard, G. Problem Definition: Tasks and Techniques. J of Sys Mgmt. 16-21; 1983 June.

267. Murphy, BG; Engel, LA. Models of the Pressure-Volume Relationship of the Human Lung. Respir Physiol. 32: 183-194; 1978.

268. Murtagh, PS; Proctor, DF; Permutt, S; Kelly, BL; Evering, S. Bronchial mechanics in excised dog lobes. J Appl Physiol. 31(3): 403-408; 1971.

269. Myhre, LG. Imposed Ventilatory Resistance. IN: Oxygen Transport to Human Tissues; Eds. Loeppky, JA; Riedesel, JA. New York, NY: Elsevier North Holland, Inc.; 267-277; 1982.

270. Neubauer, JC; Dixon, JP; Herndon, CM. Fatal Pulmonary Decompression Sickness: A Case Report. Aviat Space Environ Med. 59: 1181-1184; 1988.

271. Nielson, D; Olsen, DB. The Role of Alveolar Recruitment and De-Recruitment in Pressure-Volume Hysteresis in Lungs. Respir Physiol. 32: 63-77; 1978.

272. Nixon, W; Pack, A. Effect of altered gas diffusivity on alveolar gas exchange - a theoretical study. J Appl Physiol. 48(1): 147-153; 1980.

273. Olson, DE; Dart, GA; Filley, GF. Pressure drop and fluid flow regime of air inspired into the human lung. J Appl Physiol. 28(4): 482-494; 1970.

274. Olson, RM; Cooke, JP. Effects of Long-Hose Breathing. Aviat Space Environ Med. 49(2): 371-374; 1978.

275. Olszowka, AJ; Rahn, H. Gas Store Changes During Repetitive Breath-Hold Diving (Chapter 3). IN: Man in Stressful Environments - Diving, Hyper- and Hypobaric Physiology, Eds. Shiraki, K; Yousef, MK. Springfield, IL: Charles C. Thomas Publisher; 1987: 41-56.

276. Olszowka, AJ; Farhi, LE. A System of Digital Computer Subroutines for Blood Gas Calculations. Respir Physiol. 4: 270-280; 1968.

277. Otis, AB; McKerrow, CB; Bartlett, RA; Mead, J; McIlroy, MB; Selverstone, NJ; Radford Jr, EP. Mechanical Factors in Distribution of Pulmonary Ventilation. J Appl Physiol. 8: 427-443; 1956 January.

278. Otis, AB; Rahn, H; Epstein, MA; Fenn, WO. Performance as Related to Composition of Alveolar Air. Am J Physiol. 146: 207-221; 1946.

279. Otis, AB. The work of breathing. IN: Handbook of Physiology, Section 3: Respiration. Eds. Fenn, WO; Rahn, H. Baltimore, MD: Williams & Wilkins; 2(Chap. 17): 463-476; 1964-65.

280. Pack, A; Hooper, MB; Nixon, W; Taylor, JC. A Computational Model of Pulmonary Gas Transport Incorporating Effective Diffusion. Respir Physiol. 29: 101-124; 1977.

281. Paiva, M. Computation of the Boundary Conditions for Diffusion in the Human Lung. Comput Biomed Res. 5: 585-595; 1972.

282. Paiva, M. Gas transport in the human lung. J Appl Physiol. 35(3): 401-410; 1973.

283. Paiva, M; Lacquet, LM; Van Der Linden, LP. Gas transport in a model derived from Hansen-Ampaya anatomical data of the human lung. J Appl Physiol. 41(1): 115-119; 1976.

284. Paiva, M; Engel, LA. Model Analysis of Intra-Acinar Gas Exchange. Respir Physiol. 62: 257-272; 1985.

285. Paiva, M; Engel, LA. Pulmonary Interdependence of Gas Transport. J Appl Physiol. 47(2): 296-305; 1979.

286. Paiva, M; Yernault, JC; Van Eerdeweghe, P; Englert, M. A Sigmoid Model of the Static Volume-Pressure Curve of Human Lung. Respir Physiol. 23: 317-323; 1975.

287. Pardaens, J; Van De Woestijne, KP. Influence of Lung Inflation on the Elastic Properties of Intra- and Extrapulmonary Airways in Man. Respir Physiol. 37: 255-272; 1979.

288. Pardaens, J; Van De Woestijne, KP; Clement, J. A physical model of expiration. J Appl Physiol. 33(4): 479-490; 1972.

289. Pardaens, J; Van De Woestijne, KP; Clement, J. Simulation of regional lung emptying during slow and forced expirations. J Appl Physiol. 39(2): 191-198; 1975.

290. Pare, PD; Boucher, R; Michoud, MC; Hogg, JC. Static lung mechanics of intact and excised rhesus monkey lungs and lobes. J Appl Physiol. 44(4): 547-552; 1978.

291. Parks, DMD; Larsen, RW; Ultman, JS. Int.r Gas Mixing in the Upper and Central Airways of Man. Respir Physiol. 62: 3?5-324; 1985.

292. Paterson, DJ; Pinnington, H; Pearce, AR; Morton, AR. Maximal Exercise Cardiorespiratory Responses of Men and Women During Acute Exposure to Hypoxia. Aviat Space Environ Med. 58: 243-247; 1987.

293. Pedersen, OF; Ingram Jr, RH. Configuration of maximum expiratory flow-volume curve: model experiments with physiological implications. J Appl Physiol. 58(4): 1305-1313; 1985.

294. Pedley, TJ; Schroter, RC; Sudlow, MF. Energy Losses and Pressure Drop in Models of Human Airways. Respir Physiol. 9(3): 371-386; 1970.

295. Pedley, TJ; Sudlow, MF; Milic-Emili, J. A Non-Linear Theory of the Distrubition of Pulmonary Ventilation. Respir Physiol. 15: 1-38; 1972.

296. Pedley, TJ; Schroter, RC; Sudlow, MF. The Prediction of Pressure Drop and Variation of Resistance within the Human Bronchial Airways. Respir Physiol. 9(3): 387-403; 1970.

297. Peterman, BF; Longtin, A. Multicompartment Model of Lung Dynamics. Comput Biomed Res. 17(6): 580-589; 1984 December.

298. Peters, RM. The Energy Cost (Work) of Breathing. The Mechanical Basis of Respiration; an approach to respiratory pathophysiology. Boston: Little, Brown; Chapter 9: 195-211; 1969.

299. Petrini, MF. Distribution of Ventilation and Perfusion: A Teaching Model. Comput Biol Med. 16(6): 431-444; 1986.

300. Piwinski, S; Cassingham, R; Mills, J; Sippo, A; Mitchell, R; Jenkins, E. Decompression Sickness Incidence over 63 Months of Hypobaric Chamber Operation. Aviat Space Environ Med. 57: 1097-1011; 1986.

301. Poon, CS; Lin, SL; Knudson, OB. Optimization Character of Inspiratory Neural Drive. J Appl Physiol. 72(5): 2005-2017; 1992 May.

302. Powers, SK; Lawler, J; Thompson, D; Beadle, R. Measurement of Oxygen Uptake in the Non-Steady-State. Aviat Space Environ Med. 58: 323-327; 1987.

303. Quandieu, P; Briane, M; Henry, J; Liebaert, P. Modelisation De La Fonction Diastolique Sous Acceleration A Fort Jolt, II) Exploitation Biomecanique. Medecine Aeronautique et Spatiale. 29(115): 200; 1990.

304. Ratajczak, M. Altitude Chamber Testing of a Parachutist's High Altitude Oxygen Supply (PHAOS) System. Proceedings of the SAFE 25th Annual Symposium; Nov. 16-19, 1987; Las Vegas, Nevada. A89-10481. Newhall, CA: Safe Association. 243-248; 1987.

305. Rideout, VC; Dick, DE. Difference-Differential Equations for Fluid Flow in Distensible Tubes. IEEE Trans. Bio-Med Engr. 14(4): 171-177; 1967.

306. Riley, RL; Cournand, A. Analysis of Factors Affecting Partial Pressures of Oxygen and Carbon Dioxide in Gas and Blood of Lungs: Theory. J Appl Physiol. 4: 77-101; 1951 August.

307. Riley, RL; Permutt, S; Said, S; Godfrey, M; Cheng, TO; Howell, JBL; Shepard, RH. Effect of posture on pulmonary dead space in man. J Appl Physiol. 14(3): 339-344; 1959.

308. Robertson, PC; Anthonisen, NR; Ross, D. Effect of inspiratory flow rate on regional distribution of inspired gas. J Appl Physiol. 26(4): 438-443; 1969.

309. Rodbard, S. Recurrence of Decompression Sickness on Reascent to High Altitudes. The Air Surgeon's Bulletin. 6-7; 1944.

310. Rosburgh, HL; Ernsting, J. The Physiology of Pressure Suits. Aviation Medicine. 260-271; 1957 June.

311. Roy, R; Powers Jr SR; Kimball, W. Estimation of Respiratory Parameters by the Method of Covariance Ratios. Comput Biomed Res. 7: 21-39; 1974.

312. Sagawa, S; Shiraki, K; Miki, K; Tajima, F. Cardiovascular Responses to Upright Tilt at a Simulated Altitude of 3,700 m in Men. Aviat Space Environ Med. 64: 219-223; 1993.

313. Saidel, GM; Militano, TC; Chester, EH. Mass-Balance Model of Pulmonary Oxygen Transport. IEEE Trans Biomed Eng. BME-19(3): 205-213; 1972.

314. Saidel, GM; Militano, TC; Chester, EH. Mass-Balance Model of Pulmonary Oxygen Transport. IEEE Trans Biomed Eng. 19(3): 205-213; 1972 May.

315. Salazar, E; Knowles, JH. An analysis of pressure-volume characteristics of the lungs. J Appl Physiol. 19(1): 97-104; 1964.

316. Salmon, RB; Primiano Jr, FP; Saidel, GM; Niewoehner, DE. Human lung pressure-volume relationships: alveolar collapse and airway closure. J Appl Physiol. 51(2): 353-362; 1981.

317. Sasaki, H; Takishima, T; Sasaki, T. Influence of lung parenchvma on dynamic bronchial collapsibility of excised dog lungs. J Appl Physiol. 42(5): 699-705; 1977.

318. Sasaki, H; Hoppin Jr, FG; Takishima, T. Peribronchial pressure in excised dog lungs. J Appl Physiol. 45(6): 858-869; 1978.

319. Schall, DG. Non-Ejection Cervical Spine Injuries Due to +Gz in High Performance Aircraft. Aviat Space Environ Med. 60: 445-456; 1989.

320. Scherer, PW; Shendalman, LH; Greene, NM. Simultaneous Diffusion and Convection in Single Breath Lung Washout. Bull Math Biophys. 34: 393-412; 1972.

321. Scherer, PW; Gobran, S; Aukburg, SJ; Baumgardner, JE; Bartkowski, R; Neufeld, GR. Numerical and Experimental Study of Steady-State CO2 and Inert Gas Washout. J Appl Physiol. 64(3): 1022-1029; 1988 March.

322. Schiesser, WE. The Numerical Method of Lines. San Diego: Academic Press Inc.; 1991.

323. Schonfeld, SA; Ploysongsang, Y. Airway Closure and Trapped Gas During Low Volume Breathing. Respir Physiol. 51: 63-77; 1983.

324. Schweitzer, TW; Fitzgerald, JW; Bowden, JA; Lynne-Davies, P. Spectral analysis of human inspiratory diaphragmatic electromyograms. J Appl Physiol. 46(1): 152-165; 1979.

325. Scott, WR; Taulbee, DB. Aerosol Deposition Along the Vertical Axis of the Lung. J Aerosol Sci. 16(4): 323-333; 1985.

326. Scott, WR; Van Liew, HD. Measurement of lung emptying patterns during slow exhalations. J Appl Physiol. 55(6): 1818-1824; 1983.

327. Severinghaus, JW; Naifeh, KH. Accuracy of Response of Six Pulse Oximeters to Profound Hypoxia. Anesthesiology. 67: 551-558; 1987.

328. Shaffstall, RM; Burton, RR. Evaluation of Assisted Positive-Pressure Breathing on +Gz Tolerance. Aviat Space Environ Med. 50(8): 820-824; 1979.

329. Shapiro, AG. Steady flow in collapsible tubes. J Biomechanical Eng. 126-147: 1977.

330. Sharp, GR; Patrick, GA. Inspiratory Airflow Patterns in Subjects at Rest and During Speech With and Without Added Expiratory Resistance. Royal Air Force Institute of Aviation Medicine; IAM Report #457. 1-19; 1969 January.

331. Sharp, GR. Physiological Considerations in the Prevention of Instability in Oxygen Breathing Systems. Proceedings of AGARD Conference No. 61, AGARD CP-61-70. 1970 September.

332. Sharp, GR; Patrick, CA; Withey, WR. A Review of the Literature Relating to the Energy Expended by Pilots Flying Various Types of Aircraft. Aircrew Equipment Group Report #173; Royal Air Force Institute of Aviation Medicine, Farnborough, Hants, UK. 1-23; 1971 June.

333. Shephard, RJ. Changes of Intramuscular Blood Flow During Continuous High Pressure Breathing. Aviation Medicine. 142-153; 1957 April.

334. Sherrill, DL; Dietrich, BH; Swanson, GD. On the Estimation of Pulmonary Blood Flow from CO2 Production Time Series. Comput Biomed Res. 21: 503-511; 1988.

335. Shubrooks JR, SJ. Positive-pressure breathing as a protective technique during +Gz acceleration. J Appl Physiol. 35(2): 294-298; 1973.

336. Sikand, R; Cerretelli, P; Farhi, LE. Effects of Va and Va/Q distribution and of time on the alveolar plateau. J Appl Physiol. 21(4): 1331-1337; 1966.

337. Silverman, L; Lee, G; Plotkin, T; Sawyers, LA; Yancey, AR. Air Flow Measurements on Human Subjects With and Without Respiratory Resistance at Several Work Rates. IN: A.M.A. Archives of Industrial Hygiene and Occupational Medicine. Ed. Drinker, P. Chicago, IL: American Medical Association. 3: 461-475; 1951.

338. Slutsky, AS; Drazen, JM; O'Cain, CF; Ingram Jr, RH. Alveolar pressure-airflow characteristics in humans breathing air, He-O2, and SF6-O2. J Appl Physiol. 51(4): 1033-1037; 1981.

339. Slutsky, AS; Berdine, GG; Drazen, JM. Steady flow in a model of human central airways. J Appl Physiol. 49(3): 417-423; 1980.

340. Snyder, MF; Rideout, VC. Computer Simulation of the Venous Circulation. IEEE Trans. Bio-Med Engr. 16(4): 325-334; 1969.

341. Sobin, SS; Fung, YC; Tremer, HM; Rosenquist, TH. Elasticity of the pulmonary alveolar microvascular sheet in the cat. Circ Res. 30: 440-450; 1972.

342. Sobin, SS; Tremer, HM; Fung, YC. The morphometric basis of the sheet-flow concept of the pulmonary alveolar microcirculation in the cat. Circ Res. 26: 397-414; 1970.

343. Staub, NC. Lung Structure and Function - 1982. Basics of RD. New York, NY: American Thoracic Society (the medical section of the American Lung Association). 10(4): 1-6; 1982 March.

344. Stugging, DG; Pengelly, LD; Morse, JLC; Jones, NL. Pulmonary mechanics during exercise in normal males. J Appl Physiol. 49(3): 506-510; 1980.

345. Stuhmiller, J; Chan, F; Masiello, P; Tani, K. Test Planning, Collection and Analysis of Pressure Data Resulting from Army Weapoh Systems. Vol. II, Modelling of Far Field Data. AD-A118 290; DAMD17-78-C-8087. 69 pp; 1980 April.

346. Sudlow, MF; Olson, DE; Schroter, RC. Fluid Mechanics of Bronchial Air-Flow. Inhaled Part. 1: 19-31; 1970.

347. Surrell, WM. Pressure Flow Characteristics of Type P/Q Pressure Breathing Oxygen Masks During Steady and Dynamic Flow Conditions. Royal Air Force Institute of Aviation Medicine; Aircrew Equipment Group Report #133; Farnborough Hampshire. 1-31; 1971 January.

348. Susskind, H; Atkins, HL; Klopper, JF; Ansari, AN; Richards, P. Comparison of airway closure measured in vivo and from single-breath washout curve. J Appl Pysiol. 50(3): 587-596; 1981.

349. Sutherland, PW; Katsura, T; Milic-Emili, J. Previous volume history of the lung and regional distribution of gas. J Appl Physiol. 25(5): 566-574; 1968.

350. Taylor, G. Dispersion of soluble matter in solvent flowing slowly through a tube. Proc R Soc (London) A: Mathematical & Physical Sciences. 219: 186-203; 1953.

351. Taylor, G. The dispersion of matter in turbulent flow through a pipe. Proc R Soc (London) A: Mathematical & Physical Sciences. 223: 446-468; 1954.

352. Taylor, SJ. Estimating the Variances of Autocorrelations Calculated from Financial Time Series. Appl Statist. 33(3): 300-308; 1984.

353. Tedor, JB; Clink, JP. Man Rating the B-1B Molecular Sieve Oxygen Generation System. Human Systems Division (AFSC), Brooks AFB, TX. USAFSAM-TR-87-4. 1987 August.

354. Tesch, PA. Anaerobic Testing - Research Basis. NSCA Journal (National Strength & Conditioning Assn.); 66-69; 1984 October-November.

355. Tisi, GM; Minh, VD; Friedman, PJ. In vivo dimensional response of airways of different size to transpulmonary pressure. J Appl Physiol. 39(1): 23-29; 1975.

356. Turner, JM; Mead, J; Wohl, ME. Elasticity of human lungs in relation to age. J Appl Physiol. 25(6): 664-671; 1968.

357. Uhl, RR; Lewis, FJ. Digital Computer Calculation of Human Pulmonary Mechanics Using a Least Squares Fit Technique. Comput Biomed Res. 7: 489-495; 1974.

358. Ultman, JS; Blatman, HS. Longitudinal Mixing in Pulmonary Airways, Analysis of Inert Gas Dispersion in Symmetric Tube Network Models. Respir Physiol. 30: 349-367; 1977.

359. Van Liew, HD; Thalmann, ED; Sponholtz, DK. Diffusion-dependence of pulmonary gas mixing at 5.5 and 9.5 ATA. Undersea Biomed Res. 6(3): 251-258; 1979.

360. Van Liew, HD; Thalmann, ED; Sponholtz, DK. Diffusive Gas Mising in the Lung in Hyperbaric Environments. Prog Resp Res. 16: 110-114; 1981.

361. Van Patten, RE. Advances in Anti-G Valve Technology: What's in the Future? Aviat Space Environ Med. 59(1): 32-35; 1988.

362. Vann, RD - Ed/Chrmn. The Physiological Basis of Decompression. Proceedings of the Thirty-Eighth Undersea and Hyperbaric Medical Society Workshop, Duke University Medical Center, Durham, NC. UHMS Publication Number 75(Phys) 6/1/89; 1989 June.

363. Vann, RD; Torre-Bueno, JR. A Theoretical Method for Selecting Space Craft and Space Suit Atmospheres. Aviat Space Environ Med. 55: 1097-1102; 1984.

364. Vawter, DL; Matthews, FL; West, JB. Effect of shape and size of lung and chest wall on stresses in the lung. J Appl Physiol. 39(1): 9-17; 1975.

365. Verbanck, S; Pava, M. Model Simulations of Gas Mixing and Ventilation Distribution in the Human Lung. J Appl Physiol. 69(6): 2269-2279; 1990 December.

366. Verhamme, M; Roelandts, J; De Roo, M; Demedts, M. Gravity dependence of phases III, IV, and V in single-breath washout curves. J Appl Physiol. 54(4): 887-895; 1983.

367. Vidal Melo, MF; Loeppky, JA; Caprihan, A; Luft, UC. Alveolar Ventilation to Perfusion Heterogeneity and Diffusion Impairment in a Mathematical Model of Gas Exchange. Comput Biomed Res. 26(2): 103-120; 1993 April.

368. Vincent, NJ; Knudson, R; Leith, DE; Macklem, PT; Mead, J. Factors influencing pulmonary resistance. J Appl Physiol. 29(2): 236-243; 1970.

369. Ward, ME; Ward, JW; Macklem, PT. Analysis of Human Chest Wall Motion Using a Two-Compartment Rib Cage Model. J Appl Physiol. 72(4): 1338-1347; 1992 April.

370. Weathersby, PK; Homer, LD; Flynn, ET. On the likelihood of decompression sickness. J Appl Physiol. 57(3): 815-825; 1984.

371. Weibel, ER. Morphometry of the Human Lungs. Berlin: Springer-Verlag; 1963.

372. West, JB. Distortion of the Lung within the Chest. Fed Proc; 38(1): 11-16; 1979 January.

373. West, JB. Respiratory Physiology - the essentials, 3rd Ed. Baltimore, MD: Williams & Wilkins; 1985.

374. West, JB; Dollery, CT. Distribution of blood flow and ventilation-perfusion ratio in the lung, measured with radioactive CO2. J Appl Physiol. 15(3): 405-410; 1960.

375. West, JB; Dollery, CT; Naimark, A. Distribution of blood flow in isolated lung; relation to vascular and alveolar pressures. J Appl Physio. 19(4): 713-724; 1964.

376. West, JB; Dollery, CT. Distribution of blood flow and the pressure-flow relations of the whole lung. J Appl Physiol. 20(2): 175-183; 1965.

377. West, JB; Hugh-Jones, P. Patterns of gas flow in the upper bronchial tree. J Appl Physiol. 14(5): 753-759; 1959.

378. Whinnery, JE. Converging Research on +Gz-Induced Loss of Consciousness. Aviat Space Environ Med. 59(1): 9-11; 1988.

379. White, RJ; Croston, RC; Fitzjerrell, DG. Cardiovascular Modelling: Simulating the Human Response to Exercise, Lower Body Negative Pressure, Zero Gravity and Clinical Conditions. Adv Cardiovasc Phys. 5(1): 195-229; 1983.

380. Whitehorn, WV; Lein, A; Edelmann, A. The General Tolerance and Cardiovascular Responses of Animals to Explosive Decompression. Am J Physiol. 146: 289-298; 1946.

381. Wiggs, BR; Moreno, R/Hogg, JC; Hilliam, C; Pare, PD. A Model of the Mechanics of Airway Narrowing. J Appl Physiol. 69(3): 849-860; 1990 September.

382. Williams, JV; Tierney, DF; Parker, HR. Surface forces in the lung, atelectasis, and transpulmonary pressure. J Appl Physiol. 21(3): 819-827; 1966.

383. Wilson, TA; Bachofen, H. A model for mechanical structure of the alveolar duct. J Appl Physiol. 52(4): 1064-1070; 1982.

384. Wilson, TA. Modeling the effect of axial bronchial tension on expiratory flow. J Appl Physiol. 45(5): 659-665; 1978

385. Wilson, TA. Relations among recoil pressure, surface area, and surface tension in the lung. J Appl Physiol. 50(5): 921-926; 1981.

386. Wirjosemito, SA; Touhey, JE; Workman, WT. Type II Altitude Decompression Sickness (DCS): U.S. Air Force Experience with 133 Cases. Aviat Space Environ Med. 60: 256-262; 1989.

387. Wjote. KT; Morin, LME. Anti-G Straining Maneuver Incompatibility with Tactical Aircraft Oxygen Systems. Aviat Space Environ Med. 59: 176-177; 1988.

388. Wodicka, GR; Aguirre, A; Shannon, DC. Spectral Characteristics of Accoustic Transmission in the Thorax. IEEE/Ninth Annual Conference of the Engineering in Medicine and Biology Society. 2050-2051; 1987.

389. Wolf, CW; Petzl, DH; Seidl, G; Burghuber, OC. A Case of Decompression Sickness in a Commercial Pilot. Aviat Space Environ Med. 60: 990-993; 1989.

390. Wood, LDH; Engel, LA; Griffin, P; Despas, P; Macklem, PT. Effect of gas physical properties and flow on lower pulmonary resistance. J Appl Physiol. 41(2): 234-244; 1976.

391. Wood, LDH; Bryan, AC; Bau, SK; Weng, TR; Levison, H. Effect of increased gas density on pulmonary gas exchange in man. J Appl Physiol. 41(2): 206-210; 1976.

392. Womersley, JR. An elastic tube theory of pulse transmission and oscillatory flow in mammalian arteries. Wright Air Development Report WADC-TR-56-614. 1957.

393. Wright, CS; Clink, JP. A Study of A-37 Aircraft Oxygen Consumption. Human Systems Division (AFSC), Brooks AFB, TX. USAFSAM-TR-88-1. 1988 May.

394. Wyman, RJ. Neural Generation of the Breathing Rhythm. Ann Rev Physiol. 39: 417-448; 1977.

395. Xing, HC; Cochrane, JE; Yamamoto, Y; Hughson, RL. Frequency Domain Analysis of Ventilation and Gas Exchange Kinetics in Hypoxic Exercise. J Appl Physiol. 71(6): 2394-2401; 1991 December.

396. Yeates, DB; Aspin, N. A Mathematical Description of the Airways of the Human Lungs. Respir Physiol. 32: 91-104; 1978.

397. Yu, JS; Starkman, ES. Transient Dispersion for Mass Transport in Laminar Flow Through Tubes. The American Society of Mechanical Engineers. Presented at the Applied Mechanics and Fluids Engineering Conference; June 20-22 1973; Atlanta, GA, Fluids Engineering Division Paper #73-FE-10. 1-4; 1973.

398. Zaleski, PJ; Holden, RD. Biomedical Aspects of Oxygen Regulator Performance: I. Dynamic Characteristics. Aviat Space Environ Med. 47(5): 495-502; 1976.

399. Zaleski, PJ; Holden, RD; Hiott, BF. Biomedical Aspects of Oxygen Regulator Performance: I. Static Characteristics. Aviat Space Environ Med. 47(5): 485-494; 1976.